Probabilistic CMOS (PCMOS) in the Nanoelectronics Regime

A Thesis
Presented to
The Academic Faculty

by

Pinar Korkmaz

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

School of Electrical and Computer Engineering
Georgia Institute of Technology
December 2007

Probabilistic CMOS (PCMOS) in the Nanoelectronics Regime

Approved by:

Professor Krishna V. Palem, Adviser
School of Electrical and
Computer Engineering
Georgia Institute of Technology, Adviser

Professor Linda S. Milor
School of Electrical and
Computer Engineering
Georgia Institute of Technology

Professor James D. Meindl
School of Electrical and
Computer Engineering
Georgia Institute of Technology

Professor Ralph K. Cavin III
The Semiconductor Research Corporation

Professor Abhijit Chatterjee
School of Electrical and
Computer Engineering
Georgia Institute of Technology

Date Approved: 23rd July 2007

This dissertation is dedicated to my mother and teacher Bedriye Korkmaz. She was a gift from God to me as much as I was a gift to her. I deeply miss the presence of her beautiful soul in this world.

ACKNOWLEDGEMENTS

Many people have contributed to the completion of this dissertation that it would be impossible to thank them all here. If I fail to mention anyone by name, please know that I am thankful for everything that you have done to help me. First, I would like to thank my adviser Prof. Krishna V. Palem for his support and guidance during my study at Georgia Tech. I am very grateful to him for establishing the foundational relationship between energy consumption and probability in the domain of thermodynamics, and making the connection to CMOS circuits. Based on his ideas and foundations, this research showed that it would be possible to sustain Moore's Law using probabilistic circuits. I would also like to thank Suresh Cheemalavagu for helping me build the initial intellectual infrastructure and the foundations of probabilistic CMOS. I thank Dr. Shekhar Y. Borkar, Dr. Vivek K. De, and Dr. Keith A. Bowman for their supports through funding and their guidance. I also thank Robert Graybill who enabled this research through a DARPA seedling in 2002, for seeing the opportunity in our research and funding us.

I owe my deepest gratitude to my dissertation committee member Prof. James D. Meindl for taking the time to follow the progress of this research and his guidance and encouragement. Given his standing in microelectronics community, I consider myself very lucky to have the benefit of meeting him often. I am very happy to have worked with my committee member Prof. Ralph K. Cavin. In the short time that I interacted with him I not only had informative discussions about dynamics of noise in a CMOS inverter, but also had the chance to know an inspiring researcher and teacher. I would also like to thank the other members of my dissertation committee, Prof. Abhijit Chatterjee and Prof. Linda S. Milor for their time and valuable suggestions. I also thank Prof. Vincent J. Mooney III for giving me the opportunity to join the graduate school and for his guidance in the first two years of my study at Georgia Tech. I offer many thanks to Dr. Bilge E. S. Akgul for her guidance, collaboration and encouragement in the last two years of my graduate study.

I would like to extend my heartfelt thanks to the past and present members of the CREST group: Dr. Rodric Rabbah, Dr. Jun Cheol Park, Kiran Puttaswamy, Balasubramanian Seshasayee, Yogesh Chobe, Subramanian Ramaswamy, Jaswanth Sreeram, and Romain Cledat for their collaboration and friendship. Many thanks are due to my colleagues and good friends Angelo W. Pereira, Ozgur Celebican, Mongkol Ekpanyapong, Jacob Minz, Ismail F. Baskaya, Tushar Kumar, and Lakshmi N. Chakrapani who have been of invaluable help during my graduate study.

Without the love and support of my family, I would not be able to meet the challenges of this research. I thank my mother Bedriye for her selfless love and infinite trust in me. I thank my sister Pelin for being strong and giving me strength during the many months we worried about, fought for and finally mourned our mother. For giving me all my loved ones, I offer my thanks to God.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

As complementary metal-oxide-semiconductor (CMOS) devices approach the nano-scale in feature size, the impact of deep submicron noise poses a challenge with noise being an impediment to reliable computing. An accompanying challenge to CMOS design is to achieve low-energy computation, which has been traditionally addressed by voltage scaling. However, the utility of voltage scaling is decreasing, as reduced voltage levels reduce noise immunity even further. To fulfill both of these competing needs, this research offers an approach that treats noise as a resource rather than as an impediment. The primary objective of this research is to develop and comprehensively characterize probabilistic CMOS (PCMOS) circuits, which can be used to build energy efficient computing platforms. The simplest circuit characterized is a PCMOS inverter (switch). An analytical model relating the energy consumption per switching (E) of this switch to its probability of correctness, p is derived. This characterization can also be used to evaluate the energy and performance savings that are achieved by PCMOS switch based computing platforms. The characterization is further extended to include the effects of power supply noise, short-circuit energy dissipation, and the frequencies at which the noise signal and the output voltage of the inverter are sampled. Primitive PCMOS gates including NAND, NOR, XOR, and full adder are also characterized in terms of their p and E. Further, a methodology is developed to find the probability of larger PCMOS circuits that are composed of the primitive PCMOS gates. Another important design criterion is the speed of a PCMOS circuit. The trade-offs between the energy, speed, and p of PCMOS circuits are also analyzed. The analysis considers a PCMOS based application that has constraints on its p, performance, and energy-delay product (EDP). The analysis is able to optimize the EDP or p of a PCMOS circuit to meet the application requirements. The sensitivity of the analysis with respect to variations in temperature, supply voltage, and threshold voltage is also considered.

CHAPTER I

INTRODUCTION

The ability to scale CMOS technology as predicted by Moore's law [132] has been one of the prominent reasons for its wide use in building complex digital VLSI circuits. However, as previously studied by Packan [153], Meindl *et al.* [126] and Borkar [23], limits on device scaling and the associated challenges in terascale integration bring the necessity of new alternatives and innovations for enabling sustained technology scaling. These innovations should address the unreliable or probabilistic behaviors of future devices due to scaling challenges such as noise [100, 149], parametric variations [24], defects [68, 169], dopant concentrations [25] and tunneling effects [197]. As the transistor dimensions reduce, these statistical behaviors become more prominent, because the number of dopant atoms that control the electrical characteristics decreases as transistors are scaled. As a result, small changes in the exact number and distribution of the atoms can cause appreciable changes in the device behavior. Among these challenges posed by statistical behaviors, noise is a fundamental limit for continued transistor scaling. In Figure 1, we project the probability of error per switching (p_e) due to capacitive coupling noise (kT/C) over the years. The projection is based on the transistor length, gate oxide thickness and supply voltage values from the 2005 edition of the ITRS roadmap [82]. As seen from the figure, at year 2018, p_e is 4.58×10^{-13}. This means that a single device operating at 60 GHz can produce 100 errors in an hour. Hence, there is a need to characterize and optimize digital VLSI circuits in the presence of noise related probabilistic behaviors.

Energy efficiency is an equally serious concern in digital VLSI design given the increasing use of mobile devices and the need to reduce packaging costs. Many researchers have proposed different ideas from the device level [162, 188] to circuit [85, 116] and architectural [142, 118] levels to decrease the energy consumption of the integrated circuits and the applications running on these integrated circuits. Designing low-energy circuits in the

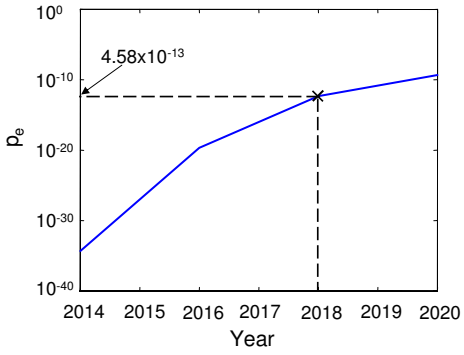
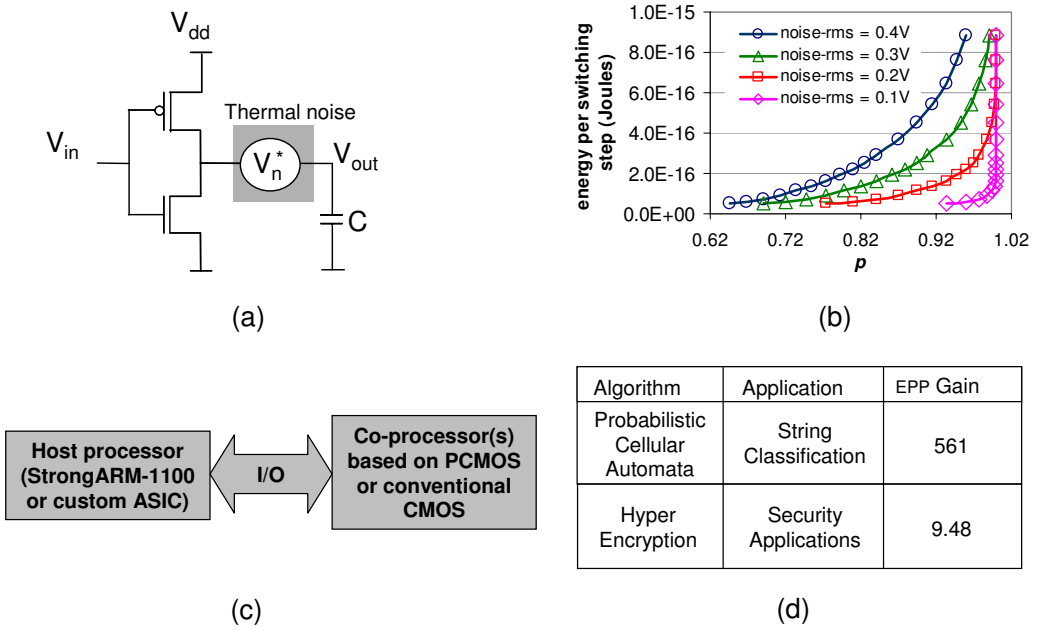


Figure 1: Evolution of the probability of error per bit switching. The error is due to the capacitive coupling (kT/C) noise.

presence of probabilistic behaviors is a challenging problem, because there is a fundamental trade-off between energy and reliability, since enhancing the robustness of a circuit introduces redundancy. For example, to cope with the probabilistic behaviors induced by noise, the supply voltage is increased, which causes an increase in the energy consumption.

In a surprising approach to meeting the challenge posed by these two competing needs and rather than treating noise as an impediment [59, 70, 115, 150], Palem [154, 155] outlined a framework for probabilistic switches and computational models based on these switches that treat noise as a resource. These models were used to show that probabilistic algorithms [135] yield low-energy computations. Thus, Palem's work established that well-characterized noise is potentially of value in realizing low-energy computing platforms for probabilistic applications based on probabilistic algorithms.

Motivated by the necessity of probabilistic approaches to digital CMOS circuits and by the usefulness of probabilistic computing substrates to achieve low-energy computing, this dissertation develops, characterizes and optimizes CMOS circuits which operate at low supply voltages and in the presence of probabilistic behaviors caused by noise. In doing so, first, the concepts of a probabilistic switch and probabilistic switching are defined. Probabilistic CMOS (PCMOS) switch, also shown in Figure 2(a) is a CMOS inverter that is coupled to noise, and whose output is correct with a probability parameter p. This switch is then characterized in detail in terms of its probability of correctness (p), and energy per switching (E). This characterization considers different types of noise couplings such as the



Algorithm	Application	EPP Gain
Probabilistic Cellular Automata	String Classification	561
Hyper Encryption	Security Applications	9.48

Figure 2: (a) A PCMOS inverter. (b) Energy-probability relationship of a PCMOS inverter realized in a 130 nm process. The root mean square (rms) value of the noise coupled to the inverter is varied from 0.4 V to 0.1 V. (c) A probabilistic system-on-a-chip (PSOC) architecture. (d) EPP gains of PCMOS-based architectures for probabilistic cellular automata (PCA) and hyper-encryption algorithms (HE). Here, EPP gain is the ratio of the EPP of the baseline implementation to the EPP of the PCMOS-based implementation. For both algorithms, baseline corresponds to the case when CMOS-based pseudo-random number generator is used to generate the random bits that are required by the algorithms.

cases when noise is coupled to the input or the power supply line of the switch. The effects of the frequency at which noise and the output voltage of the switch are sampled are also analyzed. Furthermore, the effect of the short-circuit energy dissipation is modeled. The results of this characterization are also used to quantify the energy and performance benefits gained for a subset of probabilistic algorithms implemented on probabilistic system-on-a-chip (PSOC) architectures that are realized using PCMOS inverters. Figure 2(b) depicts example results showing the energy-probability trade-offs of a PCMOS switch and Figure 2(c) shows an overview of a PSOC architecture. Figure 2(d) demonstrates the gains in the *energy* $\times$ *performance* (EPP) of probabilistic cellular automata and hyper-encryption algorithms. The trade-offs between p and E of PCMOS switches also form the basis for implementing PSOC architectures for error-tolerant DSP applications.

Based on the insights gained by characterizing a PCMOS inverter, this dissertation also

analyzes the larger CMOS gates in terms of their energy-probability trade-offs. To tackle this problem, first, the probability models for primitive PCMOS gates are developed. Then, these models are used to derive the probabilities of larger circuits. A graph-based model is used to find the probabilities of larger circuits given the probabilities of their building blocks.

Lowering the supply voltage of a PCMOS circuit decreases its dynamic energy consumption, but also decreases its p, which might have an adverse effect on the quality of solution delivered by the application. For example, an error-tolerant application would tolerate an error rate below a certain value [192]. Reducing the supply voltage also decreases the switching speed of the circuit. Therefore, to meet the performance requirement demanded by the application, the threshold voltage may have to be lowered. However, doing so would increase the static energy dissipation due to leakage. Hence, this dissertation also studies these trade-offs between the energy, performance, and p of PCMOS circuits. The analysis considers a PCMOS based application that requires its probability of correctness and performance to be in a predetermined range. By varying the supply voltage (V_{dd}) and the threshold voltage (V_{th}), the analysis optimizes the EDP of PCMOS circuits to meet the application requirements. The sensitivity of this analysis to variations in temperature, V_{th}, and V_{dd} is also considered.

1.1 Thesis Statement

Probabilistic devices and circuits that are operated in the presence of noise can be utilized to achieve low-energy computing.

1.2 Problem Statement

Low energy consumption and reliability are two competing needs in the design of VLSI systems of today and the future since improving one deteriorates the other. This research addresses the implementation of low energy probabilistic CMOS circuits that lead to significant energy savings when used in probabilistic or error-tolerant algorithms. Characterization and optimization of these probabilistic circuits in terms of their probability of

correctness, energy consumption, and performance are issues that need to be addressed.

1.3 Contributions

This dissertation presents the design and development of probabilistic CMOS circuits that are coupled to noise. The probability of correctness, energy consumption, and performance of these circuits are analyzed and optimized. The use of these probabilistic circuits to implement probabilistic and error-tolerant algorithms are also presented.

The following items are the main contributions of this research:

- **Characterization of the Energy and Probability of Correctness of a PCMOS Switch (Inverter)** An extensive characterization of the PCMOS inverter is provided. In doing so, the relationship between the probability of correctness (p) of the switch and its energy per switching (E) is established. The characterization is based on analytical models that have been developed, and later validated and supported by circuit simulations and measurements. This detailed characterization considers the following additional issues: (i) The type of the noise coupling, (ii) Short-circuit energy consumption, and (iii) The frequencies at which the noise and the output voltage is sampled. This work is the first to provide a characterization of the behavior of CMOS switches used as computing elements when their behavior is probabilistic. It thus departs from all conventional approaches to modeling switches as deterministic structures, and extends their behaviors wherein the probability of correctness is an explicit parameter. The characterization in this work, besides being of interest purely in ontological terms, also has utility since probability is becoming an explicit consideration in the design of computing systems.

- **Characterization of the Probabilistic Behavior of PCMOS Circuits**: A characterization of the probabilistic behavior of larger CMOS circuits is presented. To tackle this problem, first probability models for primitive PCMOS gates are developed. Then, a graph-based model is use to find the probabilities of larger circuits given the probabilities of their building blocks. These analytical models are validated

against circuit simulations. The effect of the logic function that a gate is implementing on its E-p characteristics is another issue that is addressed. This research provides a basis for analyzing probabilistic behaviors due to noise and other perturbations (e.g. parameter variations) in future technologies, and can be used in probabilistic design and synthesis methods to improve circuit reliability.

- **Analysis and Optimization of the Energy, Delay and Probability of PCMOS Circuits**: An investigation of the trade-offs between the energy, speed (or performance), and probability of correctness (p) of PCMOS circuits is provided. For given constraints on p, performance, and energy delay product (EDP), and using analytical models of energy, delay, and p, the optimum values of EDP and probability are found for PCMOS circuits. The analytical models are validated using circuit simulations for PCMOS circuits designed in a 0.13 μm process. The results show that, to minimize EDP, it is preferable to operate PCMOS circuits at lower supply voltages. On the other hand, to maximize p, the highest possible supply voltage under the given constraints is preferable. This analysis makes it possible to achieve an optimal circuit design that satisfies the p, performance, and EDP requirements for a given application. An analysis of the impact of variations in temperature, threshold voltage, and supply voltage on optimal EDP and probability values is also included.

- **Utilization of PCMOS Circuits in Probabilistic Computing Architectures**:

  To use noise in implementing probabilistic computing platforms, amplification of the noise may be required. This research provides a design and a detailed validation of an integrated structure that includes an amplifier that is coupled to an inverter. The resulting structure, when used in probabilistic architectures, has proved to be significantly energy efficient—even after paying the penalty for the noise amplification—compared to a conventional pseudo-random number generator based realization of probabilistic computing architectures.

1.4 Thesis Organization

The dissertation is organized into ten chapters.

- **CHAPTER I. INTRODUCTION**: This chapter provides an introduction on the probabilistic CMOS based computing. It also discusses the contributions of this research and outlines the organization of the thesis.

- **CHAPTER II. ORIGIN AND HISTORY OF THE PROBLEM**: This chapter introduces the important concepts used in this dissertation and provides an extensive survey of related work.

- **CHAPTER III. ENERGY EFFICIENT PROBABILISTIC CMOS (PCMOS) CIRCUITS AND THEIR CHARACTERISTICS**: This chapter presents a detailed characterization of a PCMOS switch (inverter) including the derivation of the analytical models for the energy and probability of the PCMOS switch, as well as the description of the framework for the circuit simulations that is used to validate the analytical models. The analytical models are also extended to include the effects of the frequencies at which noise and output voltages are sampled. The use of PCMOS switches in energy-efficient implementations of probabilistic algorithms is also outlined in this chapter.

- **CHAPTER IV.VALIDATION OF PCMOS CHARACTERISTICS USING PHYSICAL MEASUREMENTS**: This chapter describes the physical measurements that are performed to validate the energy and probability characterization of PCMOS switches. A detailed description of the experimental setup as well as a comparison of experimental results to analytical models are provided. Furthermore, this chapter presents a detailed description of the design of a thermal noise based random number generator which could be used in realizing probabilistic architectures on PCMOS-based platforms. There is also a brief discussion of the quality of the random bits that are produced by this random number generator.

- **CHAPTER V. AN IMPROVED ENERGY MODEL FOR THE PCMOS INVERTER**: This chapter discusses the impact of short-circuit energy dissipation on the energy-probability relationship of a PCMOS inverter. A detailed derivation of the short-circuit energy model of a CMOS inverter is included.

- **CHAPTER VI. PROBABILISTIC BEHAVIOR OF LARGER PCMOS CIRCUITS**: This chapter discusses the probabilistic behavior of larger PCMOS circuits. A discussion of the derivation of the analytical models for the energy-probability characteristics of primitive PCMOS gates is followed by the description of an algorithm to derive the energy-probability characteristics of PCMOS circuits that are built using primitive gates.

- **CHAPTER VII. ANALYSIS AND OPTIMIZATION OF THE ENERGY, DELAY, AND PROBABILITY OF PCMOS CIRCUITS**: This chapter considers the trade-offs between the energy, delay, and the probability of PCMOS gates. Various optimization problems and algorithms to solve these problems are also presented in this chapter. The impact of variations in temperature, supply voltage, and threshold voltage on the characteristics of PCMOS gates, as well as their optimizations is also analyzed.

- **CHAPTER VIII. REALIZING ENERGY EFFICIENT ARCHITECTURES USING PCMOS TECHNOLOGY**: This chapter discusses the use of PCMOS gates to implement probabilistic and error-tolerant applications. A discussion of the probabilistic system-on-a-chip architectures is included. This chapter also shows the utility of PCMOS in realizing signal processing applications.

- **CHAPTER IX. FUTURE DIRECTIONS FOR PCMOS RESEARCH**: This chapter explores the future directions for PCMOS research. Possible extensions to the current research are described. Following this, different switches and emerging research technologies are considered with the perspective on probabilistic computing. The chapter also briefly discusses the utility of PCMOS approach for future semiconductor technologies.

- **CHAPTER X. CONCLUSIONS**: This chapter concludes this dissertation with a summary of main contributions.

CHAPTER II

ORIGIN AND HISTORY OF THE PROBLEM

This chapter describes the important concepts used in this dissertation and summarizes the relevant prior work. Furthermore, the previous work is compared to the dissertation research.

2.1 Statistical Variations in CMOS Circuits

For several decades, the amount of information that computers are capable of processing and the rate at which they process has increased —through the scaling of CMOS transistors following Moore's Law [132], which states that transistor density and performance double every 3 years. These highly scaled devices in nanoscale CMOS [82] as well as in emerging technologies [9, 110, 237] attempting to supplement CMOS by the end of the ITRS roadmap [82], would inevitably exhibit statistical behaviors. Such behaviors are due to noise as well as variability inherent in manufacturing processes and in the operating conditions of devices.

Noise immunity becomes difficult to achieve in deep submicron (DSM) technologies due to reduction of feature sizes, smaller supply voltages (smaller noise margins), and higher density. These features render DSM technology inherently noisy with noise comprising ground bounce and IR drops [128], thermal noise [184], capacitive and inductive cross-talk [47, 217], charge leakage and charge sharing [6], and so forth. The reduction of the feature sizes leads to a decrease in the number of dopants and channel electrons in the active regions of the device. There are random fluctuations in the physical quantities of the devices such as the channel current resulting from the random variations associated with these dopants and channel electrons. From the central limit theorem, the magnitude of the random variations is inversely proportional to the number of random variables. Thus, as devices shrink, these random variations become more prominent. Sano [184] investigated

the intrinsic current fluctuations in very small Si-MOSFETs using Monte Carlo device simulations and found that the normalized standard deviation of the drain current increases as the device width is reduced to the deep submicron regime. The scaling behavior of the inductive and resistance voltage drops across the on-chip power distribution networks was studied in [128]. It was found that the signal to noise ratio (SNR) decreases by the scaling factor, S, in the case of resistive noise and by S^2 in the case of inductive noise; hence, the on-chip inductive noise increases faster and becomes more significant with technology scaling as compared to resistive noise. It was concluded that careful trade-offs between the resistance and inductance of power distribution networks in nanometer technologies will be necessary to achieve minimum power supply noise levels. It was shown in [127] that the noise immunity of dynamic digital circuits will be reduced in future technologies mainly due to the supply and threshold voltage scaling.

Parameter variations, which include the process, temperature and supply voltage variations also become more prominent with the scaling [24]. Due to limitations of the fabrication process (e.g. sub-wavelength lithography and etching) and variations in the number of dopants in the channel of short-channel devices, device parameters such as length (L), width (W), oxide thickness (T_{ox}) and threshold voltage suffer large variations [19], which in turn affect the delay and power of the circuit. With each technology generation, the area of a transistor decreases by about 50% while transistor currents decrease by only 20% to 30%. Since the voltages are no longer scaling rapidly (due to these variations), power density increases by almost 50% leading to hot spots on the die [52]. Similar to these thermal variations, supply voltages can change significantly over time due to reasons such as supply droop or overshoot events.

Given the impact of statistical variations summarized above, there is a need for new alternatives and innovations that should address these probabilistic behaviors. Next, Section 2.2 will describe various techniques that have been developed to tolerate and minimize these statistical variations and Section 2.3 will summarize the approaches that aim to achieve useful computation in the presence of these variations.

2.2 Techniques to Reduce the Magnitude and Impact of the Statistical Variations in CMOS Circuits and Systems

There has been extensive research on variation-tolerant design. One approach to designing more robust circuits is to develop new design methodologies. For example, Goel *et. al* [58] proposed a design methodology to build low-power, high-speed XOR-XNOR circuits with high noise immunity. Higher noise immunity is achieved by adding transistors to the basic designs and combining the XOR and XNOR gates with feedback transistors. Similarly, in [4] a noise-tolerant cache design was proposed. The noise tolerance was achieved by putting a diode in parallel with the gated-ground transistor—an extra NMOS transistor was introduced between the source of the NMOS transistors and the ground to improve the subthreshold leakage of a 6-transistor SRAM cell. Furthermore, various techniques have been proposed to reduce the noise due to sources such as ground bounce [10, 32, 137], cross-talk [94, 130, 148] and IR drops [36, 71]. One such technique by Henzler *et. al* [71] reduces the power supply noise of power-gated circuits by using a charge pump based activation technique. In power-gated circuits, fast block activation is desired to increase the power gain, but it causes transient peak currents which may exceed the maximum supply current. In [71], this IR-drop is reduced using a controllable block activation. In [94], an on-chip bus encoding scheme targeted for high performance generic system on a chip (SOC) is proposed. This scheme reduces the delay faults by completely eliminating the most critical type of crosstalk coupled switched capacitance. In another approach to reduce crosstalk, Badaroglu *et. al* [10] introduced a clock-skew optimization methodology that decreases simultaneous switching noise (SSN) in large digital circuits. In their solution, the design is split into different clock regions to avoid the simultaneous switching of the large circuit on the same clock edge. A common technique to reduce power supply noise of digital integrated circuit is to insert decoupling capacitors (decaps) which serve as local reservoir of current to meet the sudden current demands. There has also been substantial research effort on the placement [232, 235] and optimization [34, 108] of decoupling capacitors and their effectiveness [144, 227] in reducing the power supply noise.

To cope with parameter variations and the effects of these variations on performance

and leakage, various circuit-level techniques have been used. For example, in the forward body bias (FBB) technique [147], the device threshold voltage (V_t) is modulated for high performance by forward biasing the body. This method also reduces the short channel effects, hence reduces the V_{th} variations. Similarly, leakage power can be reduced using reverse body bias (RBB) [93], in which the PMOS substrate voltage is raised above V_{dd} and NMOS substrate is lowered below ground. These two body bias techniques were combined into adaptive body bias (ABB) [210]. Knowing that V_{th} should be lowered for the circuits that are too slow and raised for the circuits that are too leaky, ABB allows each die on a wafer (and each large circuit block on a die) to have the optimum threshold voltage that maximizes its performance subject to a power constraint. Similarly, adaptive supply voltage (ASV) [35] is used to dynamically adjust the supply voltage against the variations in supply voltage as well as the process variations, wherein the slow parts will have their operating voltages set to a higher value, whereas the faster parts will have lower supply voltages. A technique for temperature control is dynamic clock throttling [44] in which global clock is stopped from toggling to cool down to hot-spots. Moving to the microarchitectural techniques, Fetzer [52] described the use of adaptive circuits in Intel's Itanium 2 9000 series microprocessor (codenamed Montecito). In a large chip, clock skew can be up to 15% of the clock distribution's insertion delay because of the process, voltage and temperature variations. To reduce the skew, a regional active deskew (RAD) system is used in Montecito. RAD constantly monitors skew in clock distributions and makes adjustments to nullify skew.

2.3 Probabilistic Computation in the Presence of Statistical Variations

As summarized above, due to probabilistic behaviors in CMOS devices CMOS-based computing platforms are rendered probabilistic. As a result, these probabilistic computing platforms can be used for deterministic computing only when they are ameliorated with error detection and correction mechanisms. The history of studies on realizing reliable Boolean functions using noisy logic gates using hardware redundancy was first addressed by Von Neumann [224]. The construction proposed in [224] was to build reliable Boolean functions

by interleaving computational layers with error correcting layers to keep the probability of error of the overall network under control. Error correction was done using triple modular redundancy and majority voting. It was shown that a given logic function with arbitrarily high reliability can be realized by the proposed construction using noisy logic gates with an error probability of 0.0073. Elias [48] improved the reliability of computation by noisy logic gates by employing error correcting codes. Elias concluded that an arbitrarily high level of reliability can be achieved only when the computation rate approaches zero. In [166], it was shown that the depth of a reliable Boolean function implemented with noisy logic gates must be higher than that of the realization of the same with noiseless gates. However, unlike Von Neumann's work, it was not shown how the bound on the depth can be realized. In [62], it was shown that when a given logic function is implemented with 3-input logic gates, it is possible to compute reliably if the per-gate error probability is smaller than 1/6. An interesting approach to the problem of reliable computing from unreliable elements was proposed by Hegde and Shanbhag [70]. Their approach was to model the noisy logic gates and wires as noisy discrete channels. Then, using this model and the requirement that the information transfer capacity of the channel has to be greater than the information transfer rate [186] lower-bounds on the energy dissipation, circuit speed, and transition activity of digital circuits were derived. It was also shown that these lower-bounds can be approached using noise tolerance via coding. In a recent study by Nepal *et. al* [150], a novel noise-tolerant design methodology was proposed. This design technique is based on Markov Random Field (MRF)s, which are used to maximize the probability of correct state configurations of logic functions. It was shown that for CMOS circuits significant immunity to noise and threshold voltage variations can be achieved by the propagation of state variables over an MRF network. The approach is also applicable to different computing platforms such as carbon nanotube (CNT)s, quantum dot cellular automata (QCA)s and single electron transistor (SET)s.

As summarized in this section and in the former section, the efforts (such as [4, 58, 70, 150]) to cope with the challenge of uncertainty have considered noise as a source of instability and primarily aimed at yielding a degree of noise immunity (or noise tolerance) yielding

deterministic digital circuits and computing platforms. Differing from these approaches, in this dissertation, rather than being considered as a hurdle to be overcome, noise is viewed as a resource whose controlled use might in fact be useful in digital circuits. Hence, the precise relationship between the probability of correctness and the associated physical attributes (e.g. supply voltage and the amount of noise) are characterized for CMOS circuits that are rendered probabilistic by noise. In this dissertation it is also shown how these noisy circuits can be utilized to implement energy-efficient probabilistic and error-tolerant applications.

2.4 Energy Consumption of CMOS Circuits

Energy consumption of CMOS circuits consists of switching, short-circuit, and leakage energy components. Switching energy of a CMOS circuit, which we denote by E_S is due to the charging and discharging of load capacitances and internal-node capacitances:

$$E_S = \frac{1}{2} C_{eff} {V_{dd}}^2 \tag{1}$$

where C_{eff} denotes the effective switching capacitance and V_{dd} denotes the supply voltage. E_S can be reduced by decreasing C_{eff}. Effective switching capacitance can be reduced by using less logic, smaller devices, and fewer and shorter wires. Example techniques for reducing the active area include resource sharing [37, 172], logic minimization [81, 211] and gate sizing [42, 97]. The choice of logic style and circuit topology also affects the switching energy consumption. For example, the number of devices that switch per cycle can be reduced by using asynchronous circuits [14]. One common technique to reduce E_S is to reduce V_{dd}. Starting from the early work of Swanson and Meindl [205], wherein the authors developed MOS transistor models valid in the subthreshold region of operation, low voltage circuits and accurate models for the transistors of these circuits have been developed. For example, in [228], the effect of supply voltage and threshold voltage scaling on the energy and performance of a ring oscillator has been studied. Similarly, [60] has investigated the effect of reducing the supply and threshold voltages on the energy efficiency of CMOS circuits. Kuroda *et. al* [104] have described a variable supply voltage scheme, where the supply voltage is changed adaptively depending on the required frequency of operation. There are two drawbacks of reducing the supply voltage. One is the increase in the delay of gates

due to the decrease in the supply voltage. To overcome this problem, the threshold voltage is also scaled. The other drawback of reducing the supply voltage is the decreasing noise immunity of the circuits. The trade-offs between energy, performance, and noise immunity will be reviewed in Section 2.7.

Short-circuit energy dissipation (E_{sc}) occurs due to direct path currents caused by nonzero rise and/or fall times of inputs. Short-circuit current is significant when the rise/fall time at the input of a gate is much larger than the output rise/fall time. For a typical CMOS design, short-circuit energy is usually less than 10% of the dynamic energy and the ratio of the short circuit energy to dynamic energy is inversely related to the ratio of the threshold voltage to the supply voltage of the transistors [151]. There has been extensive research on modeling the short-circuit power dissipation [3, 20, 65, 151, 180, 214]. The alpha-power law MOSFET model [183] has been in the heart of a significant portion of the research on the short-circuit power [20, 101, 151, 221]. Physical MOSFET model based short-circuit dissipation models have also been developed [8, 214]. In an interesting approach to short-circuit power estimation, Acar *et. al* [3] have developed a methodology based on the timing models of CMOS gates that are used in timing analysis. Due to the increasing effect of interconnects on VLSI design, a recent short-circuit power model considers the RLC-based load models [33].

Leakage energy can be described as the energy that is dissipated through transistors without producing any useful outcome. There are three major mechanisms [181] that cause leakage energy dissipation (E_L): (1) subthreshold (2) gate (3) reverse-biased, drain- and source-substrate junction band-to-band-tunneling. Subthreshold leakage occurs because of the weak inversion conduction of a MOS transistor when the transistor's gate voltage is below the threshold voltage (V_{th}) and it increases exponentially as V_{th} increases. Reverse-biased source/drain junction leakage dissipation occurs due to the reverse-biased junction currents and depends on the junction areas and doping concentration. Gate leakage dissipation is due to the tunneling current flowing into the gate of the transistor. Gate leakage increases as gate oxide thickness (T_{ox}) is scaled down. Several leakage models for different

leakage currents such as subthreshold leakage current [91, 159], band-to-band tunneling current (BTBT) [139, 177], and gate tunneling current have been developed. In [140], authors developed accurate models for BTBT, gate and subthreshold leakage currents in logic gates such as inverter, NAND, and NOR. Rao *et. al* [176] reported an efficient technique for estimating gate leakage current by performing a logic state based analysis of the transistors. Rahman *et. al* [173] proposed a technique for fast estimation of gate and subthreshold leakage currents in digital logic circuits. Narendra *et. al* [146] developed a statistical model to estimate sub-threshold leakage based on variations in threshold voltage in the chip. In [138], the impact of loading and transistor stacking on the total leakage current was modeled.

There has also been extensive work on minimization and optimization [2, 18, 107, 111, 168, 175, 193] of leakage. At the process level, leakage reduction can be achieved by controlling the dimensions (length, oxide thickness, junction depth, etc.) and doping profile in transistors. At the circuit level, threshold voltage and leakage current of transistors can be effectively controlled by controlling the voltages of different device terminals (drain, source, gate, and substrate). For example, at the process level, supersteep retrograde wells [208] and halo implants [234] have been used as a means to scale the channel length and increase the transistor drive current without causing an increase in the OFF-state leakage [181]. The leakage reduction techniques at the circuit level include transistor stacking [88], supply voltage optimization [136], reverse body biasing [93], dynamic threshold voltage [96], multiple threshold voltage [143] and state assignment [106]. Furthermore, there are input vector control techniques [2, 64], which force the circuits into low leakage states during the standby periods.

As summarized above, decreasing V_{dd} reduces the switching, short-circuit and leakage energies. However, when V_{dd} is decreased, probability of correctness, hence the reliability of the circuit also decreases. In a surprising and new approach to countering both of these competing needs, in this dissertation, the solution that is proposed is trading of the reliability of the computation with the energy consumption of the integrated circuits. In doing so, probabilistic circuits that work at low supply voltages and in the presence of noise are developed. These probabilistic circuits are targeted to be the building blocks of energy

efficient implementations of probabilistic applications such as Bayesian inferencing [178] and probabilistic cellular automata [53], as well as DSP primitives such as fast Fourier transform (FFT) and finite impulse response (FIR) filter.

2.5 Noise Fundamentals and Noise Sources in Semiconductors

Noise is any unwanted disturbance that is interfering with a desired signal. This unwanted signal is often caused by external sources, such as the electrostatic or electromagnetic coupling between the circuit and AC power lines or the cross-talk between adjacent circuits. This type of noise can be eliminated by adequate shielding, filtering or by changing the layout of circuit components. On the other hand, the basic random mechanisms inherent in the electronic devices also cause noise, which can neither be predicted exactly, nor can be totally eliminated.

Random noise is generated by almost everything in nature. In video signals, noise appears as snow on the screen of a television set. In audio signals, noise can be heard as a background hiss. In electronic circuits, noise is controlled by careful design. Because it cannot be described by any mathematical function, it must be described by probability and statistics. Some of the important ideas of these disciplines that are applicable to noise analysis are described below.

2.5.1 Basic Concepts and Definitions

In this section, important concepts and definitions that will be helpful in understanding the properties of random signals are summarized. The reader can refer to [57, 84, 90, 158] for a detailed analysis of probability, random variables and stochastic processes.

2.5.1.1 A Brief History of Probability Theory

The classical foundation of probability theory began with the notion of equally likely classes in the eighteenth century. The classical definition of probability, formulated by Bernoulli [17] and De Moivre [131] is: the probability of an event is the ratio of the number of equally

likely cases that favor it to the total number of equally likely cases possible under the circumstances. Initially, probability theory mainly considered discrete events, and its methods were mainly combinatorial. Eventually, analytical considerations compelled the incorporation of continuous variables into the theory. This culminated in modern probability theory, the foundations of which were laid by Kolmogorov. Kolmogorov [102] combined the notion of sample space, introduced by von Mises [223], and measure theory, invented by Borel [22] and Lebesgue [67], and presented his axiom system for probability theory in 1933. This became the axiomatic basis for modern probability theory. Kolmogorov's philosophy was frequentist which views the probability of an event as the "limit" of its relative frequency in a large number of trials.

In contrast to the frequentist interpretation of probability, in Bayesian view, probability is defined as the degree to which a person believes a proposition. The Bayesian approach is named after Thomas Bayes, the originator of Bayes' theorem [13]. This theorem is often used to update the plausibility of a given statement in light of new evidence. The Bayesian interpretation of probability allows probabilities to be assigned to all propositions (or, in some formulations, to the events signified by those propositions) in any reference class of events, independent of whether the events can be interpreted as having a relative frequency in repeated trials.

Since it is possible to conduct apriori experiments and collect statistical data out of these experiments, in this dissertation, frequentist view of probability is adopted.

2.5.1.2 Random Variables and Sample Spaces

Formally, a random variable is a measurable function from a probability space into a measurable space [46]. This measurable space is the space of possible values of the variable, and it is usually taken to be the real numbers with the Borel σ-algebra [63]. Probability space is a measurable space $(\Omega, \mathcal{F}, P)$ where $(\Omega, \mathcal{F})$ is a measurable space and P is the probability measure on $\mathcal{F}$. Here, Ω is the sample space which is a nonempty set whose elements are known as outcomes, $\mathcal{F}$ is a σ-algebra of subsets of Ω, and the probability measure P is a function from $\mathcal{F}$ to the real numbers that assign to each event a probability value between

0 and 1.

In much simpler terms, a random variable is a function X that assigns to each possible outcome of an experiment a real number. The sample space of the experiment is the set of all possible outcomes. If the sample space is either finite or countably infinite, the random variable is said to be discrete. If X may assume any value in some given interval I (the interval may be bounded or unbounded), it is called a continuous random variable. In the following, simpler definitions will be provided for the sake of understandability.

2.5.1.3 Probability Density Function

Let X be a continuous real-valued random variable. A probability *density function* for X is a real-valued function f which satisfies

$$P\left(a \leq X \leq b\right) = \int_{a}^{b} f\left(x\right) dx \tag{2}$$

for all $a, b \in \Re$.

The probability density function f contains all the probability information about the experiment, since the probabilities of all events can be derived from it. In particular, the probability that the outcome of the experiment falls in an interval $[a, b]$ is given by (2), that is, by the area under the graph of the density function in the interval $[a, b]$. Hence, there is a close connection between the probabilities and areas. This connection is utilized in this dissertation too (see Section 3.2).

Similarly, in the context of calculus, the probability of occurrence of an event in the interval $[x, x + dx]$, where dx is small, is approximately given by

$$P\left[x + dx\right] \approx f\left(x\right) dx$$

that is, by the area of the rectangle under the graph of f. Note that as $dx \to 0$, the probability $\to 0$, so that the probability of a single point is 0.

Since the probability of $-\infty < x < \infty$ is 1,

$$\int_{-\infty}^{\infty} p(x) dx = 1$$

Another function that is closely related to the probability density function is the cumulative distribution function. For a continuous real-valued random variable with density function $f(x)$, cumulative distribution function is defined by

$$F(x) = \int_{-\infty}^{x} f(t)\, dt \tag{3}$$

Furthermore,

$$\frac{d}{dx}F(x) = f(x)$$

The word distribution is often used to refer to the cumulative distribution function. However, it is also used to refer to probability density function as in "normally distributed" which means that the random variable has a normal density function, which will be described below in Section 2.5.1.6. In the sequel, "distribution" and "distributed" will be used to refer to density function, whereas "distribution function" will be used to refer to cumulative distribution function.

2.5.1.4 Random Processes

As formerly described, a random variable X is a mapping from the sample space Ω to $\Re$. Similarly, a *random (stochastic) process* is a mapping from the sample space into an ensemble of time functions (known as sample functions). To every event $w \in \Omega$, there corresponds a function of time t and w, $X(t, w)$. For example, in a dice rolling experiment where the outcome is the number on the face of the dice, $\Omega = \{1, 2, 3, 4, 5, 6\}$, and one can define choose $X(t, i) = t^i$ as a random process.

For a *discrete random process*, w takes on only discrete values, whereas for a *continuous random process*, w takes on a continuum of values.

For a fixed $t = t_o$, the quantity $X(t_o, w)$ is a random variable mapping Ω to $\Re$. For fixed $w = w_o$, $X(t, w_o)$ is a well-defined, non-random, function of time. For fixed t_o and w_o, $X(t_o, w_o)$ is a real number.

For a random process, the first-order order distribution function is defined as

$$F(x, t) = P\left[X(t) \leq x\right] \tag{4}$$

The *first-order density function* is defined as

$$f(x,t) \equiv \frac{dF(x,t)}{dx} \tag{5}$$

These definitions generalize to the n^{th} order case. For any given positive integer n, let $x_1, x_2, \cdots, x_n$ denote n "realization" variables, and let $t_1, t_2, \cdots, t_n$ denote n time variables. Then, the $n^{th} - order\ distribution\ function$ is defined as

$$F\left(x_1, x_2, \cdots, x_n, t_1, t_2, \cdots, t_n\right) = P\left[X(t_1) \leq x_1, X(t_2) \leq x_2, \cdots X(t_n) \leq x_n\right] \tag{6}$$

Similarly, the $n^{th} - order\ density\ function$ is defined as

$$f\left(x_1, x_2, \cdots, x_n, t_1, t_2, \cdots, t_n\right) = \frac{\partial^n F\left(x_1, x_2, \cdots, x_n, t_1, t_2, \cdots, t_n\right)}{\partial x_1 \partial x_2 \cdots \partial x_n} \tag{7}$$

A process $X(t)$ is stationary if its statistical properties do not change with time. Hence, if

$$F\left(x_1, x_2, \cdots, x_n, t_1, t_2, \cdots, t_n\right) = F\left(x_1, x_2, \cdots, x_n, t_1 + c, t_2 + c, \cdots, t_n + c\right)$$

for all orders n and all time shifts c, then the process is stationary.

2.5.1.5 Statistical and Temporal Averages

Statistical averages are useful in describing random variables and random processes. To describe random variables, ensemble (or statistical) averages, the averages over a series of events, are required; whereas for random processes, temporal averages, that is the averages over time, are also necessary. Below, first the ensemble averages for a random variable, then the time averages of a random process are described.

Expectation of a Random Variable. Let X denote a random variable with density function $f(x)$, the *expected value* (or *mean* or *average*) of X is defined as

$$\mu = E\left[X\right] = \int_{-\infty}^{\infty} x f(x) dx \tag{8}$$

If the random variable X is discrete and takes on the value of x_i with probability $P\left[X = x_i\right] = p_i$, then the expected value of X is

$$\mu = E\left[X\right] = \sum_i x_i p_i \tag{9}$$

Variance and Standard Deviation of a Random Variable. The variance of random variable X is defined as

$$\sigma^2 = Var\left[X\right] = E\left[\left(X-\mu\right)^2\right] = \int_{-\infty}^{\infty}\left(x-\mu\right)^2 f(x)dx \tag{10}$$

The square root of the variance is called the *standard deviation* of the random variable, and it is denoted as σ. Variance is a measure of the dispersion about the mean.

The *root mean square* (*rms*) of a variable X is defined as

$$rms = \sqrt{E\left[X^2\right]} \tag{11}$$

The rms and standard deviation of a random variable are related through

$$rms^2 = \sigma^2 + \mu^2 \tag{12}$$

Hence, when the mean of a random variable is 0, its rms and σ are identical.

Moments of a Random Variable. The n^{th} moment of a random variable X is defined as

$$m_n = E\left[X^n\right] = \int_{-\infty}^{\infty} x^n f(x)dx \tag{13}$$

Note that variance can also be expressed as

$$\sigma^2 = m_2 - \mu^2 \tag{14}$$

Statistical and Time Averages of Random Processes. The expected value (or mean) of a random process $X(t)$ is defined as

$$\mu(t) = E\left[X(t)\right] = \int_{-\infty}^{\infty} x f(x,t)dx \tag{15}$$

The expected value is a first-order statistic since it depends only on the first-order density function.

The variance of a random process is defined as

$$\sigma^2\left(t\right) = E\left[\left(X\left(t\right)-\mu\left(t\right)\right)^2\right] = \int_{-\infty}^{\infty}\left(x-\mu(t)\right)^2 f\left(x,t\right)dx \tag{16}$$

On the other hand, the autocorrelation function of $X(t)$ is defined as

$$R(t_1,t_2) = E\left[X(t_1)X(t_2)\right] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_1 x_2 f(x_1,x_2,t_1,t_2)dx_1 dx_2 \tag{17}$$

Since the autocorrelation function depends on the second-order density function of $X(t)$, it is a second-order statistics.

When $X(t)$ is stationary, the mean

$$\mu(t) = E\left[X(t)\right] = \int_{-\infty}^{\infty} xf(x)dx \tag{18}$$

is constant over time, and the autocorrelation function is

$$R(\tau) = E\left[X(t)X(t+\tau)\right] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_1x_2f(x_1, x_2, \tau)dx_1dx_2 \tag{19}$$

depends only on the time difference $\tau = t_1 - t_2$.

A process $X(t)$ is *wide-sense stationary* (WSS) if its (1) mean $\mu(t) = E\left[X(t)\right]$ is constant, and (2) its autocorrelation $R(\tau) = E\left[X(t)X(t\tau)\right]$ depends only on the time difference τ.

A process is said to be *ergodic* if all orders of statistical and time averages are interchangeable. Since, the ensemble averages can not change over time, ergodicity also implies stationarity. When the process is ergodic, then the statistical average of the process is identical to its time average:

$$E\left[X(t)\right] = \langle X\left(t\right)\rangle = \frac{1}{T}\int_0^T X\left(t\right)dt \tag{20}$$

Similarly, for an ergodic random process, the correlation function is

$$R\left(\tau\right) = E\left[X(t)X(t+\tau\right] = \langle X\left(t\right)X\left(t+\tau\right)\rangle = lim_{\tau\to\infty}\frac{1}{T}\int_0^T X\left(t\right)X\left(t+\tau\right)dt \tag{21}$$

At $\tau = 0$,

$$E\left[X^2(t)\right] = \sigma^2 = \frac{1}{T}\int_0^T X^2\left(t\right)dt \tag{22}$$

2.5.1.6 Normal Probability Density Function

The normal (also referred to as Gaussian) probability density function is the most important probability function in the study of noise. If the variation in a process is caused by a large number N of random and unrelated occurrences, it can be shown that the probability density function for the process tends to a Gaussian function in the limit as $N \to \infty$. This is a statement of what is known as the Central Limit Theorem [90]. Because electronic noise

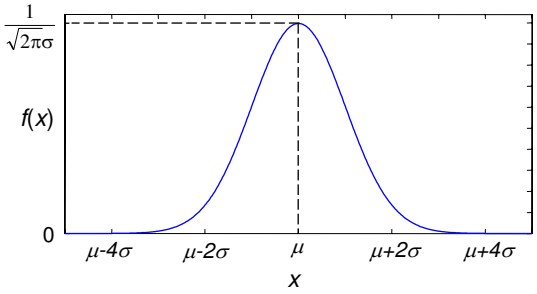


Figure 3: Normal probability density function.

is generated by the randomness in the flow of many current carriers, it can be described by the normal density function.

Let μ, $-\infty < \mu < \infty$, and σ, $\sigma > 0$, be constants. Then

$$f(X) = \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{(-X-\mu)^2}{2\sigma^2} \tag{23}$$

is a normal probability density function, where X denotes a random variable, μ and σ denote the mean and variance of this variable respectively. It is often called the bell curve because the graph of its probability density resembles a bell as shown in Figure 3. The maximum value occurs at $x = \mu$ and it is inversely proportional to σ.

2.5.1.7 Power Spectral Density

Let $X(t)$ be a WSS random process. $X(t)$ has and average power of $E[|X(t)|]$, a constant. This total average power is distributed over some range of frequencies. This distribution over frequency is referred to as power spectral density (or power spectrum), and denoted by $S_X(w)$. $S_X(w)$ is non-negative ($S_X(w) \geq 0$). The area under S_X is proportional to the average power in $X(t)$

$$\text{Average Power in } X(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_X(w) dw \tag{24}$$

Also, note that the average power is measured in Watts and $S_X(w)$ is measured in Watts/Hz.

Since, $S_X(w)$ describes the power distribution in the frequency domain, it is calculated using Fourier transform

$$S_X(w) = limit_{T\to\infty} \left[\frac{E\left[|F_{X_T}(w)|^2\right]}{2T} \right] \tag{25}$$

where $F_{X_T}(w)$ is the Fourier transform of a the truncated process $X_T(t)$ which is defined as

$$X_T(t) = \begin{cases} X(t), & |t| \leq T \\ 0 & |t| > T \end{cases} \tag{26}$$

Spectral density is related to the autocorrelation as stated by the Wiener-Khinchin Theorem [129]: *the Fourier transform of the autocorrelation is the power spectral density.*

$$S_X(w) = F[R(\tau)] \tag{27}$$

2.5.1.8 Additive White Gaussian Noise

Additive white Gaussian noise (AWGN) is a generally accepted model [43] for thermal noise in communications channels. Noise is additive, that is, the received signal is equal to the addition of the noise to the transmitted signal, wherein noise is statistically independent of the signal. Furthermore, noise has a white spectrum [109], that is, noise is a random process with a flat (constant) spectral density, which also means that the autocorrelation of noise in time domain is zero for any non-zero time offset. Finally, noise has a normal or Gaussian probability density function. Also, note that an infinite-bandwidth white noise signal is purely a theoretical construction. By having power at all frequencies, the total power of such a signal is infinite. In practice, a signal can be "white" with a flat spectrum over a defined frequency band.

In this dissertation, bandlimited AWGN noise sources are used, since many of the inherent noise sources in semiconductors produce white Gaussian noise (will be seen in Section 2.5.2) and every electronic system has a finite bandwidth.

2.5.2 Noise Mechanisms in Semiconductor Devices

The fluctuating current and voltages due to the underlying physical mechanisms in semiconductors give rise to noise. The primary types of semiconductor noise are thermal noise, shot noise, and flicker noise.

2.5.2.1 Thermal Noise

The statistical nature of scattering of free charge carriers cause random changes in their velocities and hence give rise to random microscopic currents. As a result, a fluctuating current equal to the sum of these microscopic currents exist in semiconductors along with a fluctuating voltage that can be measured at the electrical contacts. This fluctuating voltage is called thermal noise under thermal equilibrium conditions. Thermal noise is also called Johnson noise as it was first measured by Johnson [87] in 1928. A little later in the same year, Nyquist [152] derived a formulation for the mean square value thermal noise voltage across a resistor, which is given by

$$\overline{v_t^2} = 4kTR\delta f \tag{28}$$

where k is the Boltzmann's constant, T is the absolute temperature, R is the resistance and δf is the bandwidth in Hz over which the noise is measured. This simple equation is valid only as long as $f << kT/h$, where h is Planck's constant. Otherwise the mean-square thermal noise voltage is given by

$$\overline{v_t^2} = \frac{4hfR\delta f}{exp\left(\frac{hf}{kT}\right) - 1} \tag{29}$$

From the Central Limit Theorem, the amplitude distribution of thermal noise is Gaussian [216].

2.5.2.2 Shot Noise

Shot noise is generated by the random emission of electrons or by the random passage of electrons and holes across a potential barrier. The mean-square shot noise current in the frequency band δf is given by

$$i_{sh}^2 = 2qI_{DC}\delta f \tag{30}$$

where q is the electronic charge and I_{DC} is the DC current flowing through the device. This equation was derived by Schottky in 1918 [185] and is known as the Schottky formula. The spectral density of shot noise is given by

$$S_I(f) = \frac{i_{sh}^2}{\delta f} = 2qI_{DC} \tag{31}$$

The spectral density is constant with frequency; thus shot noise is white noise. Furthermore, the amplitude distribution of shot noise can be modeled by a normal or Gaussian distribution.

2.5.2.3 Flicker (1/f) Noise

1/f noise takes its name from its spectral density which is proportional to $1/f^n$, where n is usually 1, but can take values from 0.8 to 1.3 in various devices. The frequency range of 1/f noise is from frequencies as low as 10^{-6} to 10kHz [76]. The mechanisms causing 1/f noise have not been fully understood yet. There are different explanations about its origin and different formulations have been proposed for its calculation. For example, in resistive materials, its origin seems to be fluctuation of the mobility of the free charge carriers.

1/f noise is a quite common phenomenon [134]. It is observed in diodes, transistors, resistors, thermistors, thin films, and light sources. Even the fluctuations of a membrane potential in a biological system have been reported to have 1/f noise. The theory of flicker noise in semiconducting filaments was developed by McWhorter [122] in 1955. The origin of 1/f noise in MOS transistors has been the subject of great study and controversy [76, 218]. In 90's, the studies of small area sub-micron MOS transistors [79, 99] have, to a large extent, resolved the controversy and there is now a generally agreed upon physical model for 1/f noise [31, 79, 83].

Flicker noise is modeled by a noise current source in parallel with a device. In general, the mean-square flicker noise current in the frequency band δf is given by

$$i_f^2 = \frac{K_f I_{DC}^m \delta f}{f^n} \tag{32}$$

where I_{DC} is the dc current, $n \simeq 1$, K_f is the flicker noise coefficient, and m is the flicker noise exponent. K_f and m change from material to material and from technology to technology. The spectral density of 1/f noise is given by

$$S_i(f) = \frac{i_f^2}{\delta f} = K_f I_{DC}^m f^n \tag{33}$$

2.6 Probabilistic Switching and Energy-Probability Tradeoffs

In this section, we will first describe the concepts of probabilistic switch and switching. Following this, we will present a review of the research on the probability and the switching energy of such switches.

We define a *switch* as a device for realizing computations that are functions of a single bit. To realize computations, *switching* is used to alter the current bit or value, say 0, to some other value, say 1. A probabilistic switch produces the desired value as an output that is 0 or 1 with probability p, and, hence, can produce the wrong output value with a probability of $(1 - p)$. In contrast with a probabilistic switch, a conventional deterministic switch produces an output whose value is always correct.

The notion of a value such as 0 or 1 being modeled in a physical system with a single molecule dates back to 1929 and can be attributed to Szilard [206]. In particular, his work and that of several subsequent physicists was motivated by a need to explain the celebrated Maxwell's demon [206] paradox. Subsequent authors credit Szilard with having envisioned the modern notion of a "bit" and a machine with two "states".

The energy characteristics of switching have their roots in thermodynamics, the history of which, traces back to the works of Carnot [28] and Clausius [39] in the early nineteenth century, leading to significant developments in the early part of the twentieth century. Influenced significantly by Maxwell ([120, 121]), the current statistical interpretation of thermodynamics was first introduced by Boltzmann [21] and was later developed by Gibbs [56] and Planck [167].

While other celebrated researchers, including von Neumann [225], observed that the minimum energy needed to compute a bit is $kTln2$ joules, it was Landauer [105] who took a very big step toward clarifying the Maxwell's demon paradox in his widely known work. In doing so, he also explicitly laid the foundations for the (more) modern field of the thermodynamics of computation. Bennett [16] pioneered logically reversible computations, leading to the widely known models for reversible computing that admit computations with energy recovery. By contrast, in this research switching is based on nonrecovering modes

of energy consumption and computation energy once expended by a switching step is not recovered, even if such a recovery is physically feasible through reversible thermodynamic processes from a physical standpoint.

In the context of probabilistic switching, Palem [155] characterized switching as well as its associated energy consumption based on physical realizations characterized by (classical) statistical thermodynamics. This statistical foundation served as a basis here for determining the energy consumption of computations constructed from networks of switches. These switches are inherently probabilistic; they do not need an explicit random source, typically realized as a pseudo-random number generator central to the earlier development of the theory of probabilistic algorithms (see [219] for example).

Moving toward realizations of "electrical switches"—transistors, based on which modern computers are built the inherent energy needed by deterministic switching were studied by Meindl [123], and Meindl and Davis [125] wherein they established fundamental limits and derived energy lower-bounds for CMOS-based switches. A former analysis on energy consumption-reliability problem and associated limits for digital circuits was due to Stein [198], wherein he derived the relationship between the error probability and energy consumption of an inverter and showed that an error rate of 10^{-19} necessitates an energy consumption of 165kT. Similarly, Natori and Sano [149] derived a minimum energy consumption-reliability relationship for practical electrical circuits and concluded that the request on reliability of the total logic system will establish the lower limit to the downsizing of devices, and the lower limit of device size in CMOS LSI lies around 10-20 nm. Kish [100] also studied the potential technological difficulties of device scaling due to thermal noise and predicted that serious miniaturization problems may be expected in 6 to 10 years when the feature sizes decrease below 40 nm.

Hegde and Shanbhag [70] found energy lower-bounds for digital circuits operating in the presence of noise. In their work, lower-bounds on circuit speed, transition activity, dynamic energy dissipation and total energy dissipation are derived using information-theoretic concepts. Abbas, Ikeda and Asada [1] investigated the noise immunity of the static CMOS low power design schemes in terms of the logic and delay errors caused by different

kinds of noise existing in the digital CMOS circuits and concluded that dual threshold voltage scheme surpasses the other schemes from the noise immunity point of view.

In this dissertation, the simplest CMOS circuit that is characterized in terms of its energy consumption and probability is an inverter-based switch that is similar to the switch elements used by Stein [198], Natori and Sano [149], and Kish [100]. Differing from the previous work, this dissertation presents a detailed characterization of this probabilistic inverter (switch) that considers different noise couplings, the effects of the frequency at which noise and the output are sampled and short-circuit energy dissipation. The probabilistic inverter and larger probabilistic circuits (NAND, XOR, full adder for example) were analyzed and optimized in terms of their energy consumption, performance and probability of correctness.

A consistent theme in all of the previous work is that computation and, hence, the value of a bit being computed is deterministic—since computation, starting with Turing, was considered to be essentially a deterministic activity—and, thus, traditionally, its physical instantiation has not been subjected to a statistical interpretation. Furthermore, statistical variations such as noise are undesirable and the scaling limits of CMOS devices are determined based on the inherent noise sources in the devices. In contrast, this dissertation emphasizes probabilistic computation with a probability parameter p and aims to show that probabilistic circuits using noise in a controllable manner can be used to achieve low energy computing at different levels of VLSI design from circuits to architectures and to applications.

2.7 Energy, Delay and Probability Trade-offs

With the continued scaling of technology, the design constraints for integrated circuits have experienced a major change. In the past, the amount of functionality that could be integrated on chip was limited by area; today, power dissipation is the primary limiting factor. Focusing primarily on performance for high-speed circuits will result in too much power dissipation. Focusing only on energy (for example for mobile applications) is equally

inadequate, since this approach rarely achieves the required performance. Hence, the desirable optimization should either minimize energy consumption subject to a throughput constraint, or maximize the amount of computation for a given energy budget. Meanwhile, noise immunity has become difficult to achieve in deep submicron (DSM) devices due to reduced feature sizes, smaller supply voltages (hence smaller noise margins), and higher integration density, and designing low-power integrated circuits in the presence of noise is a challenging problem because it necessitates addressing the issues of energy reduction and reliable operation simultaneously. The previous approaches to these problems will be discussed below, starting with the approaches addressing optimization of energy and performance of CMOS circuits simultaneously, and continuing with analysis and optimization of energy, performance and reliability (probability of correctness) of CMOS circuits.

The methods of optimizing the energy and delay are well explored (see [11, 26, 60, 117]). Typically, an optimum in the energy-delay space has been searched for through minimization of objective functions that combine energy and delay. Various objective functions (or metrics) have been proposed to achieve energy and delay optimizations. Gonzalez, Gordon, and Horowitz [60] introduced the EDP as a metric to evaluate the energy efficiency of CMOS circuits. Minimizing the EDP of a circuit results in a particular design point in the energy-delay space where 1% of energy can be traded off for 1% of delay. Although EDP is useful for comparison of different implementations of a design, the design optimization points targeting EDP may not correspond to an optimum under desired operating conditions. As a result, other metrics have been proposed. For example, Penzes and Martin [163] proposed the metrics in the form of $E \cdot D^n$, where energy-delay efficiency index $n \geq 0$ characterizes any feasible trade-off between energy and performance. Hofstee [74] suggested the use of energy-performance ratio, which is the ratio of percent increase in energy per operation per percent increase in performance for various possible design parameters. Markovic *et al.* [117], presented methods for efficient energy performance optimizations at the circuit and microarchitectural levels by extending the energy-performance ratio approach to a succinct sensitivity analysis. The sensitivity analysis of Markovic *et al.* [117] can also be considered as optimizing for many possible metrics in the form of $E \cdot D^n$. Therefore,

each optimal point corresponds to a different value of energy-performance ratio. In this dissertation, we limit ourselves to circuits and architectures that we can optimize at an energy-performance ratio of 1 and we use the EDP metric to show the trade-offs between the energy, performance, and p. Our goal is to find the optimal V_{dd}-V_{th} operation region for PCMOS circuits given various constraints on their performance, EDP, and p.

Moving to the trade-offs between energy, performance and probability of correctness of CMOS circuits, a notable effort was due to Hegde and Shanbhag [70]. The primary purpose of their work was to derive information-theoretic lower-bounds on the energy consumption of noisy gates. In doing this, they also considered the performance and probability of correctness of these noisy circuits as constraints. Their lower-bounds guaranteed reliable computation in the presence of noise. By contrast, this dissertation research investigates the trade-offs between p, performance, and energy, wherein p is an independent design parameter and its value does not necessarily guarantee reliable computation. The main concern of this research is not reliable computing, but being able to compute under the constraints on energy, performance and p vales that are imposed by the application.

2.8 Probabilistic Algorithms

A probabilistic (randomized) algorithm is an algorithm that uses random numbers for the choices that it makes during its computation [135]. One of the early examples of randomized algorithms was by Rabin [170], where randomization was explicitly proposed as an algorithmic method for the problems in number theory and computational geometry. Since then, many techniques to devise and analyze randomized algorithms have been proposed. Some of the areas of application of randomized algorithms [92] are fingerprinting, random sampling, random sorting, partitioning and load balancing. The growth of applications of randomized algorithms has resulted from two major benefits of randomization: speed and simplicity. For example, Freivald's technique [135] provides a simple and fast solution to verifying matrix multiplication. The matrix multiplication verification problem and Freivald's technique are as follows:

Given n x n matrices, $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, we would like to verify that $\mathbf{AB} = \mathbf{C}$. Freivald's

randomized algorithm first chooses a random vector $r \in 0, 1^n$ whose each member is chosen independently and uniformly at random from 0 to 1. Then, we can compute $x = \mathbf{B}r$, $y = \mathbf{A}x = \mathbf{AB}r$, and $z = \mathbf{C}r$ in $O(n^2)$ time. If $\mathbf{AB} = \mathbf{C}$ then $y = z$. The algorithm errs only if $\mathbf{AB} \neq \mathbf{C}$, but y and z turn out to be equal. It is shown that the probability of error is $\frac{1}{2}$. If this step is repeated k ($k << n$) times, the probability of error will decrease to $(1/2)^k$, but the algorithm will be take $O(k \cdot n^2)$ of time. This method takes less time than the obvious $O(n^3)$ time matrix multiplication algorithm and is much simpler than the fastest deterministic matrix multiplication algorithm which runs in time $O(n^{2.376})$ [135].

Due to the probabilistic steps involved, probabilistic algorithms require random bits when implemented in software or hardware. As a result, various approaches have been developed to generate random bits. For example, there are pseudorandom generator (PRNG) algorithms, which uses arithmetics to generate a sequence of numbers that approximate the properties of random numbers. For example, linear feedback shift register (LFSR) based PRNGs use a shift register whose input bit is a function of its previous state. In [38], LFSR-based hardware PRNGs were described, wherein FPGA implementations for different PRNGs were developed. Similarly, in [61], LFSR-based stream ciphers were developed. These stream ciphers were not only constructed from very simple hardware but also were power efficient. The power efficiency was achieved by utilizing a variable power supply controller that can reduce the supply voltage adaptively depending on the workload of the encryption module.

Another set of well-known PRNGs is Linear congruential generators (LCGs). LCGs are defined by the recurrence relation:

$$z_{n+1} = (A \times z_n) \, mod M \tag{34}$$

where modulus, M, is a large prime integer and the multiplier, A, is an integer in the range $2, 3, \cdots, M-1$. This sequence is initialized by choosing a value of z (z_1) from $1, 2, \cdots, M$. More information about LCGs can be found in [160], where the authors present a minimal standard LCG, discuss the implementation details and present theoretical considerations.

As the use of computer networks, mobile computing and wireless communications have

become widespread; the design of cryptographic systems to keep these communication networks secure has become increasingly important. For this unique set of probabilistic algorithms, high quality random number generators (RNGs) are required. For example, a random number generator (RNG) based on the frequency instability of a running oscillator has been described in [50]. This RNG was developed for use in cryptographic systems and was the first integrated device of its type. In [164], a thermal noise based RNG integrated circuit has been shown. Since this RNG IC was intended for use in highly secure cryptographic applications, the RNG circuit is quite complicated and includes an A/D converter, sample/hold circuitry, a trans-conductance amplifier and a current-controlled oscillator.

In this dissertation, we use PCMOS circuits to implement probabilistic algorithms and we wish to show that statistical behavior of CMOS circuits can be harnessed to achieve useful computation. The PCMOS circuits used for implementing probabilistic algorithms are not only smaller and simpler than the PRNGs described above, but also more energy efficient.

CHAPTER III

ENERGY EFFICIENT PROBABILISTIC CMOS CIRCUITS AND THEIR CHARACTERISTICS

As CMOS technology scales down into the nanometer region, the impact of deep submicron noise poses a challenge [70, 100, 149, 198]. In the 2003 International Technology Roadmap for Semiconductors (ITRS) roadmap [82], it is stated that increasing noise sensitivity has become an important issue in the design of devices, circuits, and systems due to a reduction in operating voltage by 20% per technology node. On the other hand, an accompanying challenge to CMOS design involves achieving low-energy computation, which has been traditionally addressed by voltage scaling. However, the utility of voltage scaling is decreasing [60], as reduced voltage levels also reduce noise immunity even further.

The surprising premise that noise (or randomness) can be harnessed as a resource to achieve low energy computing was validated for the first time by Palem [154, 155] who outlined a framework for probabilistic switches and computational models based on these switches. These models were used to show that probabilistic algorithms [135] yield low-energy computations. Thus, Palem's work established that well-characterized noise is potentially of value in realizing low-energy computing platforms for probabilistic applications based on probabilistic algorithms.

In this chapter, we develop and detail the device level characterizations of a probabilistic CMOS (PCMOS) switch. In doing so, we consider different types of noise couplings, as well as the effects of the frequency at which noise and the output are sampled on the probabilistic behavior. We also model the probabilistic behavior of a PCMOS switch induced by power supply noise. Specifically, we analytically model the relationship between the energy per switching E, and the probability of correctness p with different types of noise couplings and validate our models using circuit simulations. For completeness, we outline the architecture and application level benefits of PCMOS switches.

The chapter is organized as follows: Section 3.1 describes the PCMOS switch (inverter). Section 3.2 outlines the analytical models developed for the PCMOS switch. The validation of the analytical models is presented in Section 3.3. Section 3.4 describes the effects of noise and output sampling frequencies on the behavior of the PCMOS switch. The application impact is briefly discussed in Section 3.5. Finally, Section 3.6 concludes this chapter.

3.1 Basic Concept

In this section, we introduce the concept of probabilistic switching, and introduce a PCMOS inverter realization of a probabilistic switch.

3.1.1 Probabilistic switch

A *switch* is a digital device with *one input* and *one output*. The output of the switch is a function f, of the input of the switch. The act of *switching* involves the invocation of the function f, which determines the output of the switch. The act of switching takes some finite amount of time T_s. The switch and its associated switching can be either deterministic or probabilistic. Let $X(t)$ and $Y(t)$ respectively denote the input and output of a switch where t denotes time. Then, for a *deterministic* switch, $Y(t_2) = f(X(t_1))$, where f:{0,1}→{0,1} is a one-input Boolean function, t_2 is the point in time when the switching ends, and t_1 is the point in time when the switching starts. By contrast and in the context of a probabilistic switch, the output $Y(t)$ depends on the probability p as shown in (35),

$$Y\left(t_2\right) = \begin{cases} f\left(X\left(t_1\right)\right) & \text{with probability} \quad p \ \ (1/2 \ < p < 1) \\ \overline{f\left(X\left(t_1\right)\right)} & \text{with probability} \quad 1-p \end{cases} \tag{35}$$

where $\overline{f(X)}$ denotes the logical complement of the Boolean function $f(X)$.

3.1.2 PCMOS inverter realization of a probabilistic switch

Informally, a CMOS inverter is a digital gate that realizes the *complement* function. Switching in this case corresponds to the flow of the switching current through the output capacitance of the inverter. In the context of the switch described in Section 3.1.1, for a deterministic inverter, $Y(t_2) = \overline{X(t_1)}$ where X and Y denote the Boolean (henceforth referred

to as the *binary* value for convenience) values of the input and the output of the inverter, respectively. The switching time $T_s = (t_2 - t_1)$ is the propagation delay of the inverter.

For a probabilistic inverter, the output to input relationship is described by (35). Since in its physical realization the probabilistic switch is a binary device, we digitize the continuous output signal of an inverter using (36) below, where the function Y is characterized by the binary value associated with the continuous output signal V_{out}, of the inverter. (The binary input value X can be expressed in a similar manner.)

$$Y(t) = \begin{cases} 1 & \text{if } V_{out}(t) \geq V_m \\ 0 & otherwise \end{cases} \tag{36}$$

In (36), V_m denotes the midpoint voltage [215] of the CMOS inverter.

3.2 Analytical Models for the Probabilistic Behavior of a PCMOS Inverter

In this section, we will first present the analytical model characterizing the probability parameter p of a probabilistic inverter (switch), when thermal noise is coupled to its input, or to its output. Following this, in Section 3.2.2, we will model the effect of power supply noise on the probabilistic behavior of a probabilistic inverter. In Section 3.2.3, we will establish and discuss the relationship between the energy per switching step E and the probability parameter p, referred to as the E-p relationship of a probabilistic inverter.

3.2.1 Analytical Modeling of the Probabilistic Inverter with Input- and Output-Coupled Thermal Noise

For *both* the cases when noise is coupled to the input (Figure 4(a)) and to the output (Figure 4(c)) of the inverter—for succinctness and unless otherwise stated, we will refer to a probabilistic inverter as an inverter in the sequel—we use the idealization consisting of an electrical switch in series with a resistor and a capacitor as shown in Figures 4(b) and 4(d), respectively. We base this idealization of an inverter on the early work of Stein [198] and of Natori and Sano [149]. The resistor R represents the effective resistance of each transistor when it is ON [149], whereas the capacitor C is the output capacitance of the inverter. Recall

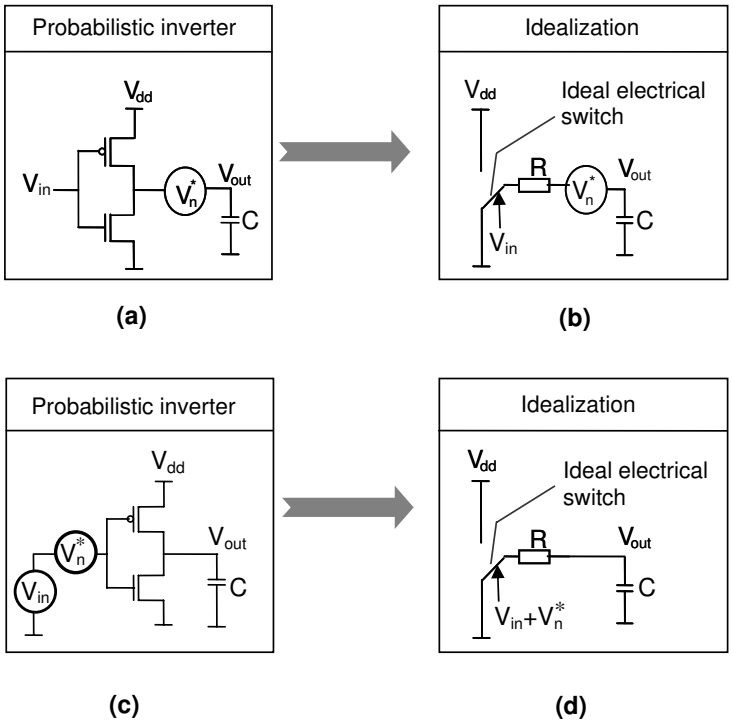


Figure 4: The idealization of a probabilistic inverter (a,b) when thermal noise is coupled to the output and (c,d) when thermal noise is coupled to the input.

that the inverter switches at a voltage V_m which corresponds to the midpoint voltage [215] of the inverter.

Following Stein [198] and Pant *et al.* [157], we consider noise sources to be random processes that are characterized by a Gaussian distribution with a standard deviation σ. Here, the value σ is referred to as the rms value of the noise. Also, in the frequency range of interest, the power spectral density of the noise is a constant, that is all of the frequency harmonics contribute equally to the magnitude.

For simplicity, in all of our studies, the sampling period of the noise is larger than the propagation delay (T_s) of the inverter, so that noise can be propagated to the output of the gate in all cases of noise coupling. Furthermore, the output voltage (V_{out}) is sampled at the same period as that at which noise is sampled. (Hence, we need not consider the (low-pass) filtering effect of the inverter on the noise.)

The behavior of a noise-induced probabilistic inverter is shown in Figure 5 and detailed in the caption. As shown there, the digital values of 0 and 1 can be altered by the noise,

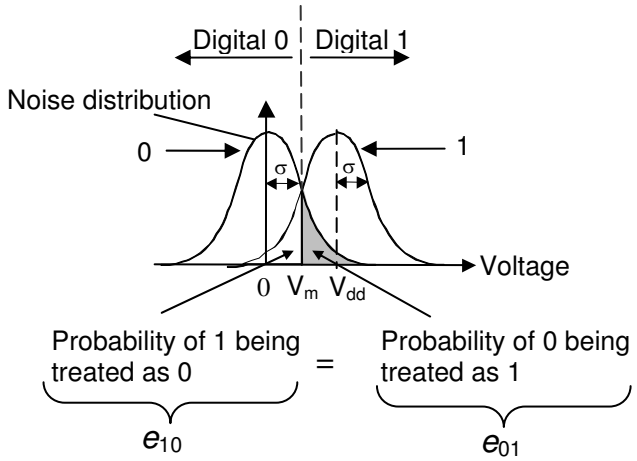


Figure 5: The digital value 0 (and 1) corresponding to the noisy output (input) voltage of the probabilistic inverter is represented by a Gaussian distribution with a mean value of 0 (or V_{dd}) and a standard deviation σ which is the rms value of the noise—modeled for both the input- as well as the output-coupled cases.

characterized by a Gaussian distribution so that a value of 0 can, with a sufficiently large noise magnitude, be sampled to be a value of 1 and vice versa. The probability of this event is determined by the area under the distribution curve corresponding to the range in question. Thus, in this figure, the probability of the output (input) digital value 0 being treated as 1, and the probability of the output (input) digital value 1 being treated as 0, are respectively, represented by the areas e_{01} and e_{10}; these correspond to the two regions in the intersection of the two noise distributions with a mean value of 0 and V_{dd} respectively. We also note that in this idealization and hence in the figure, $V_m = V_{dd}/2$, which corresponds to the special case when the transistors of the inverter are symmetric (having identical threshold voltages and satisfying the condition $(\mu_n/\mu_p) = (W/L)_p \,/\, (W/L)_n$) [215].

Given the probabilities of error, e_{01} and e_{10}, the probability of being correct, p, can be expressed as

$$p = 1 - \frac{e_{01} + e_{10}}{2} \tag{37}$$

Expressing e_{01} and e_{10} as integrals, evaluating them and substituting the results in (37) yield the following relationship between p and V_{dd}.

$$p = \frac{1}{2} + \frac{1}{4} erf\left(\frac{V_m}{\sqrt{2}\sigma}\right) + \frac{1}{4} erf\left(\frac{V_{dd} - V_m}{\sqrt{2}\sigma}\right) \tag{38}$$

In (38), erf is the error function defined as $erf(x) = 2/\sqrt{\pi}\int_0^x e^{-u^2} du$ for a real number x.

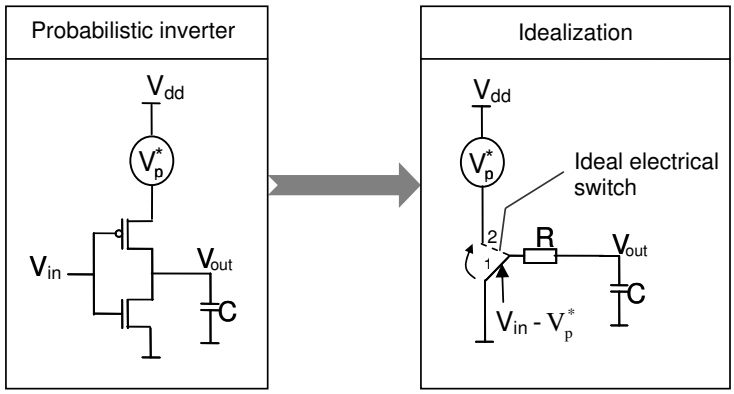


Figure 6: The approximation for a CMOS inverter coupled with power supply noise.

Note that, when $V_m = V_{dd}/2$, we find

$$p = \frac{1}{2} + \frac{1}{2} erf\left(\frac{V_{dd}}{2\sqrt{2}\sigma}\right) \tag{39}$$

which is the expression for the p of a symmetric inverter. Due to its simplicity, we will use this equation in various places in the sequel.

3.2.2 Analytical Modeling of the Probabilistic Inverter Coupled to Power Supply Noise

Following Pant *et al.* [157], power supply noise is characterized by a Gaussian distribution with rms value of σ_p. In this case, we use the idealization illustrated in Figure 6.

Referring to Figure 6(a), when $V_{in} = V_{dd}$ and if the power supply noise were not present, the NMOS transistor will be ON and the PMOS transistor will be OFF. This corresponds to a situation where the ideal electrical switch in Figure 6(b) will be at position 1. Considering the case where $V_{in} = 0$ and if the power supply noise were not present, the PMOS transistor will be ON, and the NMOS transistor will be OFF; therefore the ideal switch will be at position 2. However, due to the power supply noise, the PMOS transistor might be turned ON, even in the case where $V_{in} = V_{dd}$ and might be OFF even though $V_{in} = 0$. Considering the two cases of V_{in} (being either 0 or V_{dd}), we have determined the probability of 1 being interpreted as 0 (e_{10}) and the probability of 0 being interpreted as 1 (e_{01}) and found that the probability of being correct, p, in the case of a probabilistic inverter with power supply

noise coupling to be

$$p = \frac{1}{2} + \frac{1}{4}erf\left(\frac{V_{mp}}{\sqrt{2}\sigma_p}\right) + \frac{1}{8}erf\left(\frac{V_{dd}-V_m}{\sqrt{2}\sigma_p}\right) + \frac{1}{8}erf\left(\frac{V_{dd}-V_m}{\sqrt{2}\sigma_p}\right)\cdot erf\left(\frac{V_{mp}}{\sqrt{2}\sigma_p}\right) \quad (40)$$

The key steps of the derivations of e_{01} and e_{10}, and V_{mp} parameter of the above equation are summarized below.

1. When $V_{in} = 0$

   (a) The gate to source voltage of the PMOS transistor is

   $$V_{gsp} = -V_{dd} - V_p{}^* \quad (41)$$

   (b) From (41), the probability of $|V_{gsp}| < |V_{Tp}|$ is

   $$Pr\left(|V_{gsp}| < |V_{Tp}|\right) = \frac{1}{2} - \frac{1}{2}erf\left(\frac{V_{dd}-|V_{Tp}|}{\sqrt{2}\sigma_p}\right) \quad (42)$$

   (c) If $|V_{gsp}| < |V_{Tp}|$, the probability of V_{out} being digital 0 is $\frac{1}{2}$.

   (d) If $|V_{gsp}| \geq |V_{Tp}|$, the probability of V_{out} being digital 0 is $\left(\frac{1}{2} - \frac{1}{2}erf\left(\frac{V_m}{\sqrt{2}\sigma_p}\right)\right)$.

   Hence, the probability of 1 being interpreted as 0 (e_{10}) is

   $$e_{10} = Pr(|V_{gsp}| \geq |V_{Tp}|)\cdot\left(\frac{1}{2} - \frac{1}{2}erf\left(\frac{V_m}{\sqrt{2}\sigma_p}\right)\right) + Pr(|V_{gsp}| < |V_{Tp}|)\cdot\frac{1}{2} \quad (43)$$

2. When $V_{in} = V_{dd}$

   (a) V_{gsp} is described by

   $$V_{gsp} = V_{dd} - (V_{dd} + V_p{}^*) = -V_p{}^* \quad (44)$$

   (b) At the output, an incorrect transition to 1 can occur if $|V_{gsp}| > V_{mp}$ with

   $$V_{mp} = \sqrt{(\mu_n/\mu_p)\frac{\left(\frac{W}{L}\right)_n}{\left(\frac{W}{L}\right)_p}V_{dd}\left(\frac{3V_{dd}}{4} - V_{Tn}\right)} + |V_{Tp}| \quad (45)$$

   Hence, e_{01} is found to be

   $$e_{01} = Pr\left(V_p{}^* \geq V_{mp}\right) = \frac{1}{2} - \frac{1}{2}erf\left(\frac{V_{mp}}{\sqrt{2}\sigma_p}\right) \quad (46)$$

3. Using (37), (43) and (46), (40) is derived.

3.2.3 The E-p Relationship of a PCMOS Inverter

In this section, we will first provide analytical models characterizing the (switching) energy consumed per one switching step, denoted as E, of a PCMOS inverter. Following this, we will depict the relationship between p and E which we refer to as the E-p relationship.

3.2.3.1 Modeling the Energy Consumed by a PCMOS Inverter per Switching Step

The main components of dynamic energy consumption of a digital circuit are switching energy consumption (E_{sw}) and short-circuit energy consumption (E_{sc}). In Section 3.2.3.2, we will consider the switching energy consumption to characterize the E-p relationship of an inverter. Later, in Chapter 5, we present an improved energy model wherein the short-circuit energy consumption is also included.

3.2.3.2 Deriving the E-p Relationship for a PCMOS Inverter

We model the switching energy consumption to be

$$E_{sw} = \frac{1}{2} C V_{dd}^2 \tag{47}$$

where the energy consumed is that used to charge or discharge a capacitive load C of the RC circuit idealization of an inverter shown in Figures 4 and 6.

Recall from (38), (39) and (40) that p is a function of V_{dd}. We denote this function by h_i, where i is an integer associated with the type of coupling. Since, we have considered three cases of coupling, $i \in \{1, 2, 3\}$, where input-coupled thermal noise, output-coupled thermal noise and power supply noise are associated with indices $i = 1$, 2 and 3 respectively. Thus, for input coupled noise serving as a basis for our probabilistic inverter, $p \equiv h_1(V_{dd})$, and from (38)

$$h_1(V_{dd}) = \frac{1}{2} + \frac{1}{4} erf\left(\frac{V_m}{\sqrt{2}\sigma}\right) + \frac{1}{4} erf\left(\frac{V_{dd} - V_m}{\sqrt{2}\sigma}\right) \tag{48}$$

Equivalently, $V_{dd} = {h_1}^{-1}(p)$ where $h_i^{-1}(x)$ denotes the inverse of the function $h_i(x)$ for a real valued x. With the power supply noise coupling, on the other hand, $h_3(V_{dd})$ is

$$h_3(V_{dd}) = \frac{1}{2} + \frac{1}{4} erf\left(\frac{V_{mp}}{\sqrt{2}\sigma_p}\right) + \frac{1}{8} erf\left(\frac{V_{dd} - V_m}{\sqrt{2}\sigma_p}\right) + \frac{1}{8} erf\left(\frac{V_{dd} - V_m}{\sqrt{2}\sigma_p}\right) \cdot erf\left(\frac{V_{mp}}{\sqrt{2}\sigma_p}\right) \tag{49}$$

From (47) by substituting $h_i^{-1}(p)$ for V_{dd}, the generalized E-p relationship which is our analytical model is

$$E = \frac{1}{2} C \left[h_i^{-1}(p) \right]^2 \tag{50}$$

For example, if the transistors of the inverter are symmetric, and considering an inverter rendered probabilistic by thermal noise coupled to the input or the output of this inverter, then from (39) the analytical model characterizing the E-p relationship can be specialized to yield

$$E = 4C\sigma^2 \left[inverf\,(2p-1) \right]^2 \tag{51}$$

where *inverf* is the inverse of the error function [201].

To understand the relationships between E, p and σ, we will now elaborate on (51). Using the bounds for *erfc* derived by Ermolova and Haggman [49], we have

$$p < 1 - 0.28 e^{-1.275 \frac{{V_{dd}}^2}{8\sigma^2}} \tag{52}$$

Using this expression to lower-bound V_{dd} and hence the switching energy $E = \frac{1}{2} C {V_{dd}}^2$, we have, for a given value of p,

$$E(p, C, \sigma) > C\sigma^2 \left(\frac{4}{1.275} \right) \ln \left(\frac{0.28}{1-p} \right) \tag{53}$$

Clearly, E is a function of the capacitance C, determined by the technology generation, rms value of the noise σ and the probability of correctness p. For a fixed value of $C = \hat{C}$ and $p = \hat{p}$, $\hat{E}_{\hat{C},\hat{p}}(\sigma)$ is defined as $\hat{E}_{\hat{C},\hat{p}}(\sigma) = \hat{C}\sigma^2 \left(\frac{4}{1.275} \right) \ln \left(\frac{0.28}{1-\hat{p}} \right)$. Similarly, for fixed values of $C = \hat{C}$ and $\sigma = \hat{\sigma}$, $\hat{E}_{\hat{C},\hat{\sigma}}$ is a function of p defined as $\hat{E}_{\hat{C},\hat{\sigma}}(p) = \hat{C}\hat{\sigma}^2 \left(\frac{4}{1.275} \right) \ln \left(\frac{0.28}{1-p} \right)$.

In computer science, the notion of asymptotic complexity is widely used to study the efficiency of algorithms. Usually, efficiency is characterized by the growth of the running time (or space), of the algorithm as a function of the size of its inputs [41, 66, 171]. The O notation provides an asymptotic *upper-bound*. In this context, for a function $f(x)$ where x is from the set of natural numbers

$$f(x) = O(h(x)) \tag{54}$$

given any function $h(x)$, whenever there exist positive constants c and x_0 such that $\forall x \geq x_0$, $0 \leq f(x) \leq ch(x)$.

Similarly, the symbol Ω is used to characterize an asymptotic *lower-bound* on the rate of growth of a function. For a function $f(x)$ as before,

$$f(x) = \Omega\left(h(x)\right) \tag{55}$$

whenever there exist positive constants c and x_0 such that $\forall x \geq x_0$, $0 \leq ch(x) \leq f(x)$. In this context, the O and the Ω notation is defined for functions over the domain of natural numbers. We now extend this notion to the domain of reals. For any $y \in (\alpha, \beta)$ where $\alpha, \beta \in \{\Re^{+} \cup \}$

$$\hat{h}\left(y\right) = \Omega_{\gamma}\left(g\left(y\right)\right) \tag{56}$$

whenever there exists a $\gamma \in (\alpha, \beta)$ such that $\forall y \geq \gamma$, $0 \leq g\left(y\right) \leq \hat{h}\left(y\right)$. Intuitively, the conventional asymptotic notations capture the behavior of a function $h\left(x\right)$ "for very large" x. Our modified notion Ω_{γ} captures the behavior of a function $\hat{h}\left(y\right)$, defined in the interval (α, β). In this case, $\hat{h}\left(y\right) = \Omega_{\gamma}\left(g\left(y\right)\right)$ if there exists some point γ in the interval (α, β) beyond which $0 \leq g\left(y\right) \leq \hat{h}\left(y\right)$. This notion means "the function $\hat{h}\left(y\right)$ eventually *dominates* $g\left(y\right)$ in the interval (α, β)". We will now use this asymptotic approach to determine the rate of growth of energy described in (53), as follows.

Let us now express the lower-bound we stated in (53) using the novel asymptotic (Ω_{γ}) notation. Again fixing $C = \hat{C}$ and $\sigma = \hat{\sigma}$, let us now consider $\hat{C}\hat{\sigma}^2\left(\frac{4}{1.275}\right)\ln\left(\frac{0.28}{1-p}\right)$ from (53), and compare it against the *exponential* (in p) *function*, $E^{e}_{\hat{C},\hat{\sigma}}\left(p\right) = \hat{C}\hat{\sigma}^2 exp\left(p\right)$. We note that, when $p = 0.5$,

$$\hat{C}\hat{\sigma}^2\left(\frac{4}{1.275}\right)\ln\left(\frac{0.28}{1-p}\right) < E^{e}_{\hat{C},\hat{\sigma}}\left(p\right) \tag{57}$$

Furthermore, both functions are monotone increasing in p and they have equal values at $p \approx 0.87$. Hence,

$$\hat{C}\hat{\sigma}^2\left(\frac{4}{1.275}\right)\ln\left(\frac{0.28}{1-p}\right) < E^{e}_{\hat{C},\hat{\sigma}}\left(p\right)$$

whenever $p > 0.87$. Then from the definition of Ω_{γ}, an asymptotic lower-bound for $\hat{E}_{\hat{C},\hat{\sigma}}\left(p\right)$ in the interval $(0.5, 1)$ is

$$\hat{E}_{\hat{C},\hat{\sigma}}\left(p\right) = \Omega_{\gamma}\left(E^{e}_{\hat{C},\hat{\sigma}}\left(p\right)\right) \tag{58}$$

Let $E^{q}_{\hat{C},\hat{p}}(\sigma) = \hat{C}\left(\frac{4}{1.275}\right)\ln\left(\frac{0.28}{1-\hat{p}}\right)\sigma^2$. Referring to (53) and considering $\tilde{E}_{\hat{C},\hat{p}}(\sigma)$ for a fixed value of $C = \hat{C}$ and $p = \hat{p}$, then using Ω_γ notation, an asymptotic lower-bound for $\tilde{E}_{\hat{C},\hat{p}}(\sigma)$ is

$$\tilde{E}_{\hat{C},\hat{p}}(\sigma) = \Omega_\gamma\left(E^{q}_{\hat{C},\hat{p}}(\sigma)\right) \tag{59}$$

Then, from (58), we conclude that for any fixed technology generation (which determines the capacitance $C = \hat{C}$) and constant noise magnitude $\sigma = \hat{\sigma}$, the switching energy $\hat{E}_{\hat{C},\hat{\sigma}}$ consumed by a probabilistic switch grows with p. Furthermore, the order of growth of $\hat{E}_{\hat{C},\hat{\sigma}}$ in p is asymptotically bounded below by an exponential in p since $\hat{E}_{\hat{C},\hat{\sigma}}(p) = \Omega_\gamma\left(E^{e}_{\hat{C},\hat{\sigma}}(p)\right)$.

Similarly, from (59), we conclude that for any fixed probability $p = \hat{p}$ and a fixed technology generation (which determines the capacitance $C = \hat{C}$), $\tilde{E}_{\hat{C},\hat{p}}$ grows quadratically with σ since $\tilde{E}_{\hat{C},\hat{p}}(\sigma) = \Omega_\gamma\left(E^{q}_{\hat{C},\hat{p}}(\sigma)\right)$.

3.3 Validation of Analytical Models

To validate our analytical model of a probabilistic inverter (equation (50) from § 3.2.3.2) we performed circuit simulations in HSPICE using models of inverters realized using AMI Semiconductor (AMIS) 0.5 μm and Taiwan Semiconductor Manufacturing Company (TSMC) 0.25 μm processes. The simulation parameters are summarized in Table 1. As seen in the table, these parameters include the supply voltage (V_{dd}), the rms value of the thermal noise (σ), as well as rms value of the power supply noise (σ_p). The load capacitance value (C) corresponds to the capacitive load due to a fanout of four (which is the typical value of load used especially for delay calculations [73, 77]), leading to values of 60 fF and 28 fF for 0.5 and 0.25 μm processes, respectively. We will now detail the simulation results based on the 0.25 μm process. The results for an inverter realized using the 0.5 μm process show similar trends.

3.3.1 Modeling of Noise in Circuit Simulations

In general, a noise source in circuit simulations for transient analysis [40] is modeled as a PWL (piecewise linear) voltage source. Thus, in our simulations, noise is injected into the HSPICE "netlists" in the form of a PWL voltage source. The data points of the PWL source

Table 1: Simulation parameters for inverters in TSMC 0.25 μm and AMI 0.5 μm technologies.

Technology		**AMI 0.5 μm**	**TSMC 0.25 μm**
Inverter fan-out		4	4
Load capacitance		60 fF	28 fF
Nominal Vdd (V)		5	2.5
Transistor size	$\mathbf{(W/L)}_{pmos}$	15 μm/0.6 μm	2 μm /0.3 μm
	$\mathbf{(W/L)}_{nmos}$	6 μm/0.6 μm	0.8 μm/0.3 μm
Vdd (V)		0.5-5	0.5-2.5
σ **(V)**		0.2-0.8	0.2-0.8
σ_p **(V)**		0.2-0.8	0.2-0.8
Input rise and fall time		0.2 ns	0.2 ns

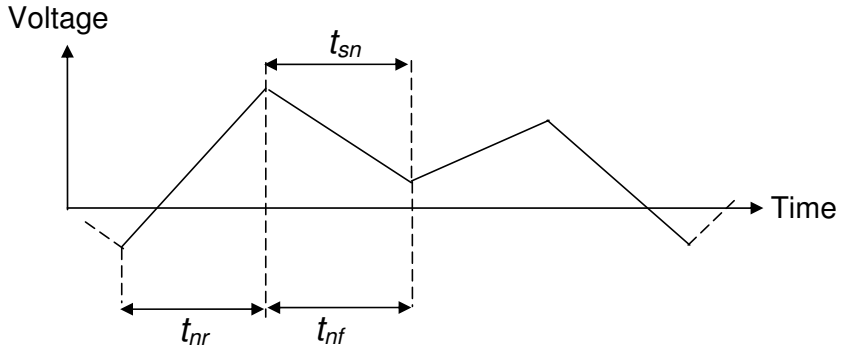


Figure 7: The noise pulse and its rise and fall times.

are derived from a Gaussian distribution of random numbers generated by Matlab. We show a sketch of an example noise pulse in Figure 7. As shown there, t_{nr} and t_{nf} denote the rise and fall times of the noise pulse, and they are identical. In addition, t_{sn} denotes the sampling period of the noise, and is equal to t_{nf} (or t_{nr}). In addition, in the current section, V_{out} is sampled with the same sampling period as that of the noise.

3.3.2 Measurement of the Energy E and the Probability p During Circuit Simulations

The energy per switching step E is determined by measuring the total current drawn from the voltage supply node of the inverter during the time that the inverter switches, where the switching is induced by a pulse source applied to its input. For completeness, the last row of Table 1 shows the value of the rise and fall times of the input pulse. We note that we only measure the energy consumed by a CMOS inverter while its output changes (switches)

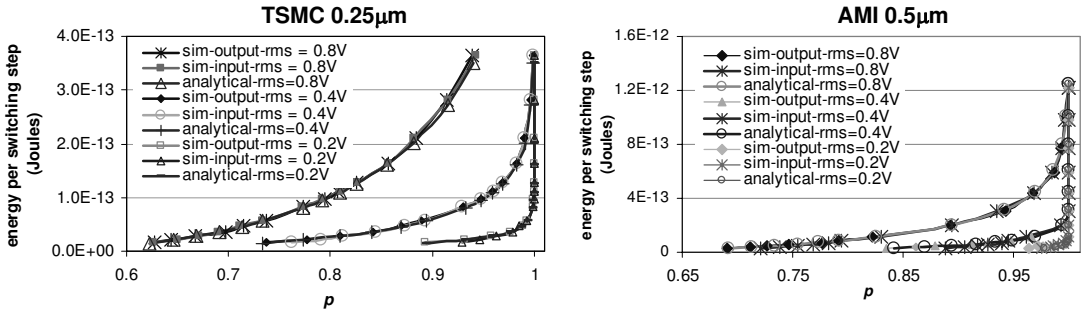


Figure 8: The E-p relationship for inverters coupled to thermal noise at their inputs or outputs.

from 0 to V_{dd} or vice versa, hence only during the act of switching.

The value of p is measured to be the ratio of the number of the correct simulation points measured at the output of the inverter to the ratio of the total number of simulation points.

We will now compare the analytical and simulation results for the E-p relationship of a probabilistic inverter that is coupled with thermal noise at its input or its output. Following this, we compare the analytical and simulation results for the E-p relationship of a PCMOS inverter coupled with power supply noise.

3.3.3 The E-p Relationship for a PCMOS Inverter Coupled to Thermal Noise

In Figure 8, we depict the E-p relationship of PCMOS inverters realized using 0.25 μm and 0.5 μm processes, both coupled to thermal noise at their outputs or inputs. The two parameters that we vary are the noise rms value σ and the operating supply voltage V_{dd}. In particular, for each value of σ, we compare values of the parameter p determined at different values of V_{dd} using the analytical model (equation (38)) with those determined using circuit simulations. In Figure 8, sim-output denotes the simulation results in the case of output-coupled thermal noise and sim-input denotes the simulation results in the case of input-coupled thermal noise. Recall that we estimate the energy consumed per switching step of the inverter analytically using (47). As shown in Figure 8, the difference between the results of analytical model and simulations is negligible. The maximum deviation between the analytically estimated and the simulated results is 3.92%, a value that can not be visually noticed. To reiterate, given a fixed amount of available noise, the energy needed

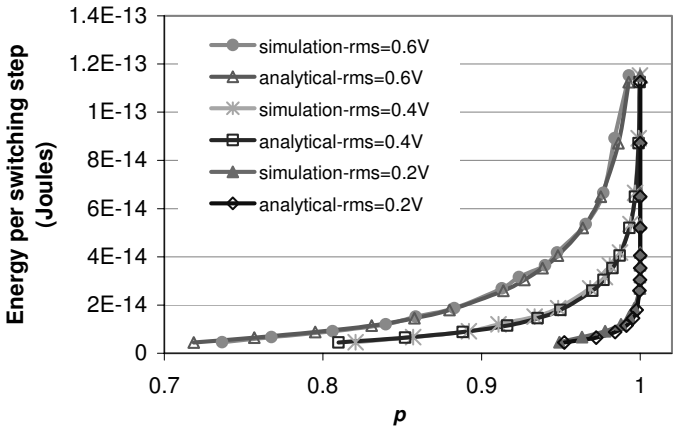


Figure 9: The E-p relationship of a PCMOS inverter with power supply noise coupling.

to produce a single bit increases with p and the rate of growth of E with p dominates an exponential function (analytically shown in Section 3.2.3.2). Furthermore, for a fixed probability value of p, the energy consumed to produce a bit increases quadratically with noise rms (also shown in Section 3.2.3.2) .

3.3.4 The E-p Relationship for a PCMOS Inverter Coupled to Power Supply Noise

The E-p relationship of an inverter coupled to power supply noise is shown in Figure 9. Again we vary the supply voltage of the inverter and the rms value of the power supply noise coupled to the inverter. The trends relating E to p, and to the noise rms value are similar to those observed in cases of the input- and output-coupled thermal noise discussed before. As seen in Figure 9, the difference between the analytical results and the simulation results is negligible and is a maximum of 3.8%.

In Figure 10, we compare the E-p relationship of an inverter coupled to thermal noise with the one coupled to power supply noise. In the legend of this figure, thermal-rms denotes the rms value of thermal noise, whereas ps-rms denotes the rms value of the power supply noise. As seen from the figure, at a fixed value of E, output-coupled thermal noise is more effective and induces a lower value of p when compared to the power supply noise of the same rms value. (See points A and B in Figure 10 for example, both of which correspond to an rms value of 0.6V and are respectively associated with thermal and power

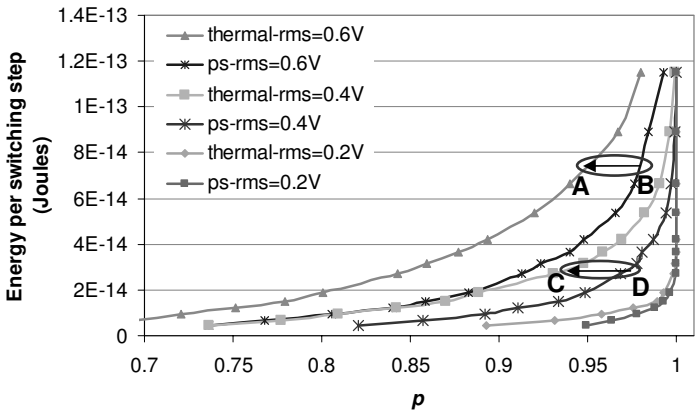


Figure 10: Comparison of the E-p relationships in the instances of power supply noise coupling and output-coupled thermal noise.

supply noise. Similarly, points marked C and D correspond to an rms value of 0.4V.) Thus, output-coupled thermal noise is more effective in realizing a specific value of p, since the required value of noise rms in the case of the output-coupled thermal noise is lower than the corresponding value for power supply noise. This is due to the fact that power supply noise induces less errors when $V_{in} = V_{dd}$; namely when the switch (in Figure 6) is at position 1 for the most part, the noise source on the supply node will be isolated.

3.4 The Impacts of Output Sampling Frequency and Noise Sampling Frequency on the Probabilistic Behavior of a PCMOS Switch

In Section 3.2, we postulated the sampling period of the noise to be larger than the propagation delay of the inverter. We also postulated that the output voltage of the inverter (V_{out}) is sampled at the same rate as the noise. In the sequel, we refer to the sampling period of the noise as t_{sn}, and the sampling period of V_{out} as t_{so}. Hence, the output sampling frequency is $1/t_{so}$ and the noise sampling frequency is $1/t_{sn}$. However, in a realistic scenario, the noise sampling period and its associated frequency may not be equal to the period of switching and its associated frequency—the reciprocal of the propagation delay is the switching frequency of the inverter. Similarly, due to a variation in the period, the output sampling frequency may not be equal to the noise sampling frequency. To comprehensively study the effects of these two frequency (or time period) parameters on the

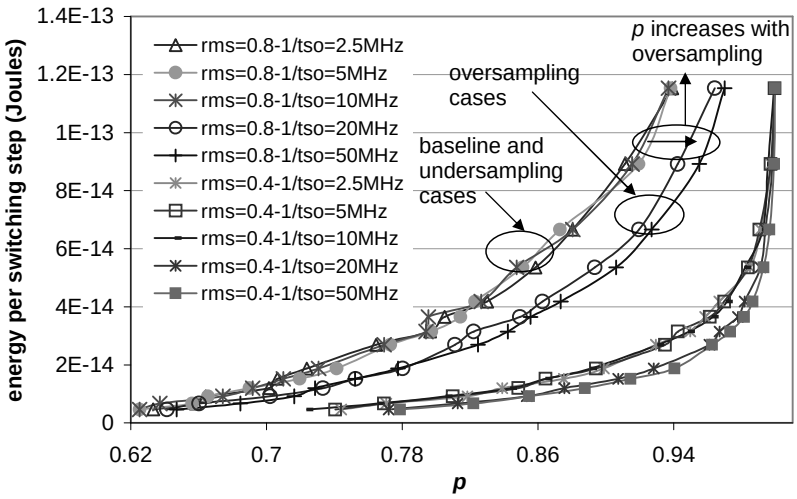


Figure 11: The impact of t_{so} on the E-p relationship

probabilistic behavior of a CMOS inverter, in Section 3.4.1 below, we will characterize the effect of varying output sampling frequency and in Section 3.4.2, that of varying the noise sampling frequency.

3.4.1 The Impact of Output Sampling Frequency on the E-p Relationship

To investigate the effects of varying the output sampling frequency, we consider oversampling (wherein the output voltage (V_{out}) of the inverter is sampled more frequently than the output-coupled thermal noise) and undersampling (wherein V_{out} is sampled less frequently than the output-coupled thermal noise). Since similar trends are observed in case of the other types of noise couplings, we limit ourselves to a discussion for the case of output-coupled noise. In this section, we consider a CMOS inverter with the parameters shown in Table 1 for our HSPICE simulations.

In Figure 11, we show the impact of oversampling and undersampling on the E-p relationship in the case of output-coupled noise. Note that the case when the output sampling frequency, $1/t_{so}$, has a value of 10 MHz—and is then equal to the sampling frequency of noise—is referred to as the baseline. The output sampling frequency ($1/t_{so}$) values of 5 and 2.5 MHz correspond to the case when output is undersampled. Similarly, the output sampling frequency values of 20 and 50 MHz correspond to the case when output is oversampled.

From the analytical model in (38) (see Section 3.2.1), p is related to the rms value (σ) of the noise. In case of oversampling, the effective value of noise rms is decreased, and thus, p is increased (at a fixed value of E). This trend can be seen from Figure 11. To show the validity of our simulation results, below, we derive p in case when the sampling frequency is $2/t_{sn}$ and we sketch how to find p for sampling frequency values other than $2/t_{sn}$.

Table 2: The variation in the average value of p across different noise rms values and output sampling frequencies

	Noise rms = 0.4 V			Noise rms = 0.8 V		
	Output sampling frequency ($1/t_{so}$)			Output sampling frequency ($1/t_{so}$)		
	20 MHz	40 MHz	100 MHz	20 MHz	40 MHz	100 MHz
p_{avg}	0.929	0.935	0.936	0.807	0.818	0.817

As seen in Figure 11, increasing sampling frequency beyond 20 MHz ($2/t_{sn}$) has a negligible impact on the E-p relationship especially at lower values of noise rms. This is also observed from Table 2, which shows the average value of p (p_{avg}) at rms values of 0.4 V and 0.8 V at different values of output sampling frequency. For example, as seen from Table 2 when rms value of the noise is 0.4 V, the difference between p values for the cases of $1/t_{so}$=20 MHz and $1/t_{so}$=40 MHz is only 0.646%, and for the cases of $1/t_{so}$ = 20 MHz and t_{so} = 100 MHz it is 0.753% on the average.

In the case of undersampling, however, the effective rms value of noise remains the same (provided that the number of output samples is large enough to preserve the original Gaussian distribution), and hence p is not affected. Therefore, we expect undersampling not to have an impact on the E-p relationship, which is validated through the simulation results shown in Figure 11.

3.4.1.1 Oversampling at a Frequency of $2/t_{sn}$

To reiterate, the noise voltage source used in HSPICE simulations is a piecewise linear voltage source, which is often used to realize transient simulations with noise [40]. Hence, when an inverter is coupled to noise at its output, its output voltage is also a piecewise linear signal. Figure 12 shows the output voltage waveform and the corresponding samples for

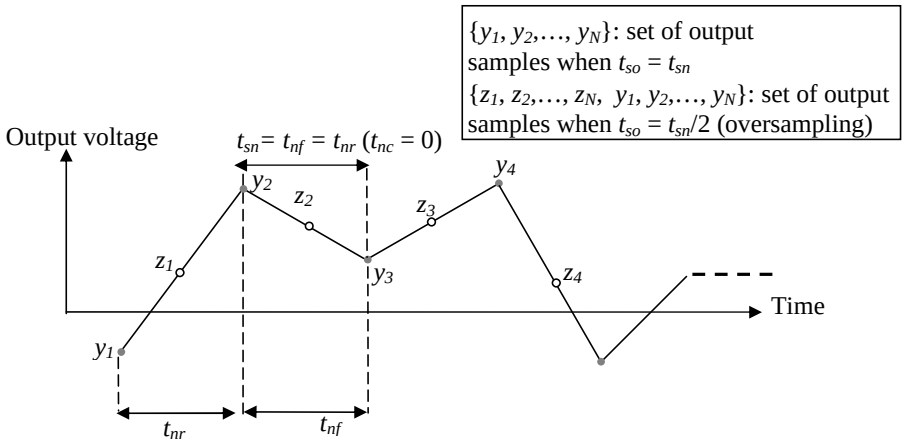


Figure 12: The output voltage waveform and the output samples with/without oversampling

the cases when the output voltage is sampled at the rate $1/t_{sn}$ (baseline) and at the rate $2/t_{sn}$ (oversampling).

In Figure 12, the samples denoted as y_i represent the output samples obtained at the rate $1/t_{sn}$, while the samples denoted by z_i represent the additional output samples obtained at the rate $2/t_{sn}$. We note that at the sampling rate of $2/t_{sn}$, the set of output samples is $z_1, z_2, \cdots, z_N, y_1, y_2, \cdots, y_N$ as shown in the figure. We denote the set of y_is by Y and the set of z_is by Z. Hence, $Y = \{y_1, y_2, y_3, \cdots, y_{n-1}, y_n\}$ and $Z = \{z_1, z_2, z_3, \cdots, z_{n-1}, z_n\}$. $C(Y)$ denotes the cardinality of Y and $C(Z)$ denotes the cardinality of Z.

To reiterate, p (obtained through simulations) is equal to the ratio of the number of the correct simulation points to the ratio of the total number of simulation points. Hence, the probability of being correct for the case when the output is sampled at the rate $1/t_{sn}$, denoted by p_{sn}, can be described as follows

$$p_{sn} = \frac{\#\text{ of correct simulation points (at sampling rate } 1/t_{sn})}{\text{total }\#\text{ of simulation points (at sampling rate } 1/t_{sn})} = \frac{C\left(Y_c\right)}{C\left(Y\right)} \tag{60}$$

where Y_c denotes the set of the correct simulation points in Y. Similarly, Z_c denotes the number of correct simulation points in Z.

Then, for the case when output is oversampled, the probability of being correct, denoted by p_{OS}, is described as follows

$$p_{OS} = \frac{\#\text{ of correct simulation points (at sampling rate } 2/t_{sn})}{\text{total }\#\text{ of simulation points (at sampling rate } 2/t_{sn})} = \frac{C\left(Y_c\right) + C\left(Z_c\right)}{C\left(Y\right) + C\left(Z\right)} \tag{61}$$

Since oversampling is realized at the rate of $2/t_{sn}$, $C(Y) = C(Z) = N$, where N denotes the total number of simulation points in the case when output is sampled at the rate of t_{sn}. Therefore,

$$p_{OS} = \frac{C\left(Y_c\right) + C\left(Z_c\right)}{2N} = \frac{1}{2}\frac{C\left(Y_c\right)}{N} + \frac{1}{2}\frac{C\left(Z_c\right)}{N} \tag{62}$$

Now, let us consider the two sets, Y and Z, each of which has a cardinality of N. The probability of being correct for set Y, denoted by p_Y, is given by the ratio $\frac{C(Y_c)}{N}$. Similarly, the probability of being correct for set Z, denoted by p_Z is given by the ratio $\frac{C(Z_c)}{N}$. Hence, when the output is oversampled at the rate of $2/t_{sn}$, from (62)

$$p_{OS} = \frac{p_Y + p_Z}{2} \tag{63}$$

Below, using the analytical model of p, which was described in Section 3.2.1, we show that p_Y and p_Z can be expressed as a function of the rms value of the noise and the supply voltage, and we derive a relationship between p_{OS} and p_{sn}.

Referring to Figure 12, when the output is sampled at the rate of $2/t_{sn}$, an element (output sample) z_i in Z, is related to two consecutive elements, y_i and y_{i+1}, of Y as follows

$$z_i = \frac{y_i + y_{i+1}}{2} \tag{64}$$

Below, we first show that the elements of Y and Z come from a Gaussian distribution. Second, we compute the rms value of the Gaussian distribution that the elements of Z are derived from.

Lemma 1 *In the case when output is sampled at the rate of $1/t_{sn}$, each element in Y comes from a Gaussian distribution with rms value of σ, wherein σ is also the rms value of the Gaussian distribution of the thermal noise coupled to the output of the inverter.*

Proof: Since the inverter is coupled to noise at its output, at a point in time, t, $V_{out}(t) = V_o(t) + {V_n}^*(t)$. Here, $V_o(t)$ denotes the potential difference between the drain nodes of the inverter transistors and the *ground* at a point in time, t. Since the output voltage is sampled at the same rate as the noise is sampled ($1/t_{sn}$), if the output is sampled at time t', $V_{out}(t')$ corresponds to the sum of ${V_n}^*(t')$ and $V_o(t')$, and this is valid for every

such t' (every such sample). Furthermore, the input voltage of the inverter is kept constant, hence $V_o(t)$ is also constant. The proof follows from the fact that for a random variable $\mathbf{x}$ characterized by a Gaussian distribution with variance σ^2, $\mathbf{x}$+C (C is constant) also has a Gaussian distribution with variance σ^2 [158]. ■

In the following lemma and proof, we consider only the case when C is zero. If C is not zero, we can define another set Z^*, such that each element of Z^* is identical to z_i-C (i is an integer in [1, N]). The following lemma and proof is also valid for Z*.

Lemma 2 *When the output voltage of the inverter is sampled at the rate of $2/t_{sn}$, each element in Y comes from a Gaussian distribution with rms value of σ and each element in Z comes from a Gaussian distribution with rms value of σ_Z, wherein σ_Z is approximately equal to $\frac{\sigma}{\sqrt{2}}$.*

Proof: Each element in Y comes from a Gaussian distribution with rms value of σ follows from Lemma 1.

From (64), each element in Z is a linear combination of the elements in Y. Therefore, each element in Z also comes from a Gaussian distribution [158]. Using (64) and the definition of the variance (see Section 2.5), the variance of the Gaussian distribution for the elements in Z, σ_Z, is described by

$$\sigma_Z = \frac{1}{N}\sqrt{\frac{y_1^2+y_2^2+y_3^2\ldots+y_n^2}{4}+\frac{y_2^2+y_3^2+y_4^2\ldots+y_{n-1}^2}{4}+\frac{2y_1y_2+2y_2y_3+\ldots+2y_{n-1}y_n}{4}} \tag{65}$$

In (65), the third ratio in the square root is equal to 0, since the elements in Y are uncorrelated. From here, Lemma 2 follows.

■

Then, from (38) (see Section 3.2), (63), and Lemma 2, we compute p_Y and p_Z and find

$$p_{OS} = \frac{1}{2}+\frac{1}{8}erf\left(\frac{V_m}{\sqrt{2}\sigma}\right)+\frac{1}{8}erf\left(\frac{V_{dd}-V_m}{\sqrt{2}\sigma}\right)+\frac{1}{8}erf\left(\frac{V_m}{\sigma}\right)+\frac{1}{8}erf\left(\frac{V_{dd}-V_m}{\sigma}\right) \tag{66}$$

Since $p_Z > p_Y$,

$$p_{OS} = \frac{p_Y+p_Z}{2} > p_Y = p_{sn} \tag{67}$$

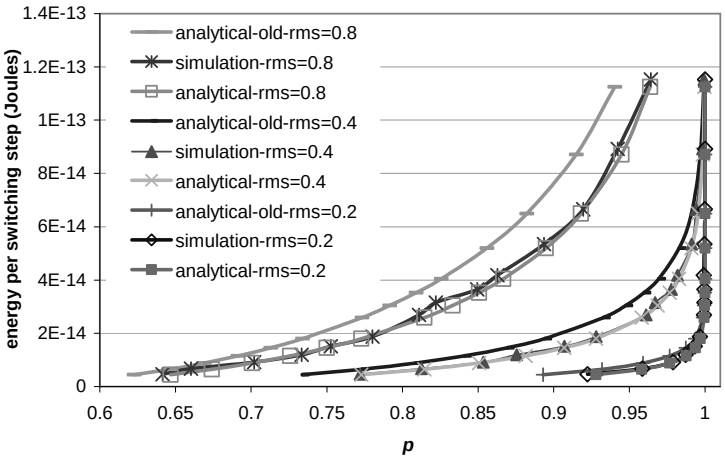


Figure 13: The *E-p* relationship in the case of the output being sampled at $2/t_{sn}$

Figure 13 shows a comparison of the analytically calculated *E-p* relationship (using (66)) and the *E-p* relationship obtained through the simulations when the output is sampled at the rate $2/t_{sn}$. In this figure, we also show the analytical *E-p* relationship obtained using (38). As seen in the figure, the results obtained using the analytical model of (66) shows a strong match with the simulation results unlike the results of the analytical model using (38) (which we refer to as analytical-old in the figure).

3.4.1.2 A Sketch of Analytical Model of p when the Output Voltage is Oversampled

In the case of when output is sampled at the rate $2/t_{sn}$, we identified the sets Y and Z. Similarly, for the other sampling rates, $1/t_{so} > 1/t_{sn}$, we can identify such sets, and find the probability of being correct for each set. We can also express p_{OS} in terms of the values of probability of being correct for these sets and show that p_{OS} is always larger than the probability of being correct for the baseline case.

As we previously showed in Figure 11, oversampling beyond $2/t_{sn}$, that is, $1/t_{so} > 2/t_{sn}$, has a negligible impact on p, especially at lower values of the rms value of the noise. Referring to Figure 11, at an rms value of 0.8V and at a fixed value of E, p increases when sampling frequency increases from 20 MHz to 50 MHz for p >0.8. However, we do not observe a steady increase in p as the frequency of sampling increases. This is because of the averaging of the probability values as described by (63). Furthermore, it can be shown that

p_{OS} will always be lower than p_Z (probability of being correct in the instance of sampling at the rate $2/t_{sn}$).

3.4.2 The Impact of the Equivalent Noise Bandwidth on the E-p Relationship

In this section, we investigate the impact of the noise sampling frequency on the E-p relationship for an inverter with input-coupled thermal noise. We only focus on the case of input coupling rather than the cases of output coupling or power supply noise coupling due to the following reasons. First, we have observed that the noise sampling frequency has negligible effect on the probabilistic behavior for the output-coupled noise. Second, our study of the effect of the noise sampling frequency for the case of power supply noise has yielded results similar to those described below for the case of input-coupled thermal noise. In this section, we will use we use 39 to compute p due to its simplicity.

Given that noise is sampled at the rate $1/t_{sn}$, the maximum frequency component of the noise should be smaller than $1/2t_{sn}$ (from Nyquist's criterion). In what follows, we will refer to $1/2t_{sn}$ as the equivalent noise bandwidth, denoted as ENBW. In the results shown below, the noise and the output are sampled at the same frequency ($S = 1/t_{sn} = 1/t_{so}$).

In practical systems, there is always a bandwidth limitation on the noise [216]. In such a bandlimited system, the rms value of the noise is proportional to the bandwidth of the system [134]. Based on this proportionality and using the alpha-power law MOSFET delay model [183], we model the effect of the filtering on the rms value of the noise and computed an equivalent noise rms value observed at the output

$$\sigma_{eq} = \sigma_i \left(\frac{K_1 T_n}{\left(K_2 T_n + K_3 \frac{V_{dd}}{(V_{dd} - V_{th})^{\alpha}} \right)} \right)^{0.5} \tag{68}$$

where σ_i is the rms value of the noise and T_n is the reciprocal of the maximum frequency component of the noise. K_1, K_2 and K_3 are empirical parameters fitted using HSPICE simulations. V_{th} and α are the threshold voltage and velocity saturation index parameters of the alpha-power law MOSFET model. Hence, when the bandlimiting effect is considered, from (39) the probability of correctness of a PCMOS inverter is

$$p = 0.5 + 0.5 erf\left(\frac{V_{dd}}{2\sqrt{2} \cdot \sigma_{eq}} \right) \tag{69}$$

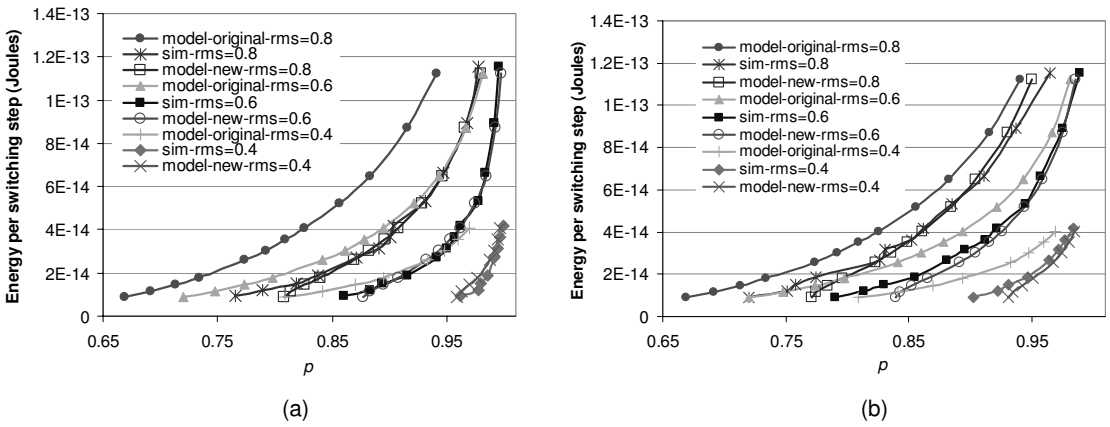


Figure 14: The effect of the filtering of the noise on the E-p relationship of a PCMOS inverter with input-coupled noise.

In Figure 14, we depict our analytical results showing the energy per switching step (E) versus p of a PCMOS inverter realized in a 0.25 μm process and coupled to noise at its input as well as the simulation results validating our analytical models. The maximum frequency component of noise is 1 GHz and 100 MHz in Figure 14(a) and (b), respectively. Different p and E values are found by varying the supply voltage (V_{dd}) of the circuit from 0.7 V to 2.5 V. The rms value of the noise is varied from 0.4 V to 0.8 V. We performed circuit simulations using HSPICE [78]. In the figure, *model-original* represents the analytical results found using (39), whereas *model-new* represents the analytical results found using (69). As seen from the figure, compared to the results corresponding to the original analytical model, the results of the model in (69) are much closer to the simulation results.

3.5 Application Impact

So far, we have presented characterization of the PCMOS switch behavior. In this section, we show the utility of such PCMOS switches and quantify the energy savings possible through using PCMOS technology in implementing one probabilistic application. We note that probabilistic methods [135] have a wide range of applications in different areas, such as pattern recognition, encryption, classification, and optimization. We wish to emphasize that the description in this section is merely meant to highlight the value and utility of PCMOS technology to probabilistic applications. Details of the architectural and application

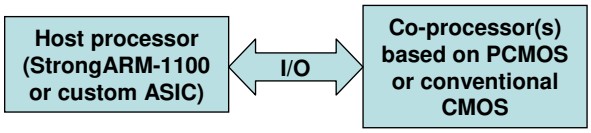


Figure 15: A probabilistic system-on-a-chip architecture.

oriented benefits of PCMOS technology will be presented in Chapter 8. A comprehensive study of this subject can be found in a publication due to Chakrapani *et al.* [29].

As we will sketch below, PCMOS devices can be used to build probabilistic system-on-a-chip (PSOC) architectures. The application- and architecture-level savings are quantified using the product of the energy consumed (measured in Joules) and the performance (measured in seconds), denoted $energy \times performance$. We present an outline of the low-energy computing PSOC architectures based on PCMOS, and the accompanying benefits in the context of the probabilistic cellular automata (PCA) application [53].

3.5.1 Probabilistic system-on-a-chip (PSOC) architectures

To realize energy efficient embedded computing platforms, we have developed a methodology for using PCMOS. As shown in Figure 15 (and discussed in detail by Chakrapani *et al.* [29]), a PSOC architecture is comprised of a host processor and a co-processor where the host processor is used to compute most of the control-intensive deterministic components of an application, whereas the co-processor realized using PCMOS can be viewed as an energy-performance accelerator that executes the probabilistic content of the application.

We compared the gain of a PSOC based implementation over a functionally equivalent SOC realized using conventional CMOS technology—both the PCMOS and CMOS designs are realized using a 0.25 μm TSMC process. The gain of the PSOC over SOC is defined as the ratio of the energy-performance product (EPP) of the SOC implementation to the EPP of PSOC implementation: $Gain = \text{EPP}_{soc}/\text{EPP}_{\text{PSOC}}$. The gain of the PSOC design over the SOC design is a factor of 774 in the context of a core primitive probabilistic operation of the PCA application (see Table 3), whereas it is a factor of 561 (not shown in the table) in the context of the overall execution of the string classification application solved using a PCA (see ref. [29] for details).

Table 3: EPP Gain of PCMOS over CMOS and over conventional software based implementation running on StrongARM SA-1100 processor to execute the primitive probabilistic operation of PCA.

Algorithm	Application	Primitive operation	Gain over CMOS	Gain over software
PCA	String Classification	Evaluating the probabilistic transition function	7.74×10^2	4.17×10^4

3.6 Conclusions

In this chapter, we have presented the results of a study of the energy-probability or E-p relationship of a PCMOS inverter. We have considered different couplings of noise, which include the input-, and output-coupled thermal noise, and power supply noise coupling. The main contribution involved the development of analytical models whose quality was validated by simulation results for all three instances of noise coupling.

Throughout, we have considered "controllable" noise sources, wherein the spectral distribution and the sampling frequency of the noise are known a priori. We have shown that sampling frequency of the noise is also critical in determining the probability parameter p associated with an inverter and developed an empirical model that captures the effect of the noise sampling frequency and the propagation delay of the inverter on p. Furthermore, the frequency at which the output of a probabilistic inverter is sampled has also been shown to affect the probability parameter p.

Our work provides analytical models and insights to the circuit designers who might wish to exploit noise in realizing probabilistic designs yielding PSOC architectures—such architectures will be more detailed in Chapter 8. Thus, using our analytical models, one can design circuits and ultra energy-performance efficient PSOC architectures as outlined in Section 3.5.

CHAPTER IV

VALIDATION OF PCMOS CHARACTERISTICS USING PHYSICAL MEASUREMENTS

To validate the energy gains resulting from using PCMOS switches we implemented PCMOS inverters in silicon. The results found from the physical measurements on these inverters are further used to quantify the energy and performance savings from using these switches in implementation of probabilistic algorithms such as Bayesian network based inference algorithm [113] and hyper-encryption [45] as we will describe later in Chapter 8.

As described in Chapter 3, the probabilistic nature of the PCMOS inverter is resulting from the noise affecting it. In Chapter 3, we considered that noise is available, and we did not account for any energy cost to produce noise. However, the availability of the noise may not be guaranteed. Even if the noise is available, it might have a small rms value (e.g., rms value of the thermal noise of a 10 MΩ resistor is only 4 mV) or the estimation of the rms value of the noise might be a difficult task to do (e.g., to estimate the power supply noise of a chip, the details about the chip should be known [157]). Thus, it is also necessary to develop other methods of generating noise or randomness. One such method is to amplify the thermal noise of a resistor [164].

We have fabricated two IC's, one in AMI 0.5 μm, another in TSMC 0.25 μm process to do hardware measurement based characterization of our probabilistic primitives and to evaluate the energy cost of the generation and amplification of the noise. The basic building blocks of these two IC's are inverters, inverter chains, subthreshold amplifiers, and random number generators built using the subthreshold amplifiers.

As we outlined in Section 3.5, PCMOS devices are used to build PSOC architectures . In realizing these architectures, the PCMOS inverters are used to generate random bits (0 or 1) with probability p. While the energy and performance gains for probabilistic applications have been our primary concern to demonstrate the utility of PCMOS, the

quality of the implementation of a probabilistic algorithm—hence, the quality of the random bits produced by PCMOS inverters, is a characteristic of interest as well. In this chapter, we employ statistical tests from the NIST suite [174] to assess to quality of randomness in a preliminary way. The random sequences in the case of PCMOS have been produced from physical measurements of a probabilistic inverter fabricated using the 0.25 μm process, whereas the pseudo-random bits derived using Park-Miller [160] algorithm were evaluated using the output of a custom design simulated using HSPICE.

The chapter is organized as follows: Section 4.1 presents an overview of the prototype chips and a detailed description of the design of the thermal noise based random number generator. Following this, in Section 4.2, an elaborate description of the framework for the measurement of the energy consumption and p of PCMOS switches is included. The measurement frameworks for the subthreshold amplifier, as well as the noise based random number generator are also explained in this section. The experimental results are discussed in Section 4.3. Finally, Section 4.4 concludes this chapter.

4.1 Overview of the Prototype Chips

Figure 16 shows the die photo of the 0.5 μm prototype chip. This chip is fabricated in AMIS CFN process. This design is synthesized into the IIT Standard Cell Library [80]. The relevant components of the chip are the inverter, inverter chain, subthreshold amplifier and the component which we refer to as the "subthreshold amplifier + inverter" in the figure. Inverter chain, subthreshold amplifier, and subthreshold amplifier + inverter are designed using Cadence Virtuoso, a custom layout tool. Gate level netlists are simulated in HSPICE and the functionality of every component is verified using these simulations. Among the relevant components of the chip, the subthreshold amplifier+inverter serves as the thermal noise based random number generator. Next, we will describe the design of this random number generator.

4.1.1 Thermal Noise Based Random Number Generator

Random number generators (RNGs) are useful for a variety of purposes such as generating data encryption keys, random initialization of certain variables in cryptographic protocols,

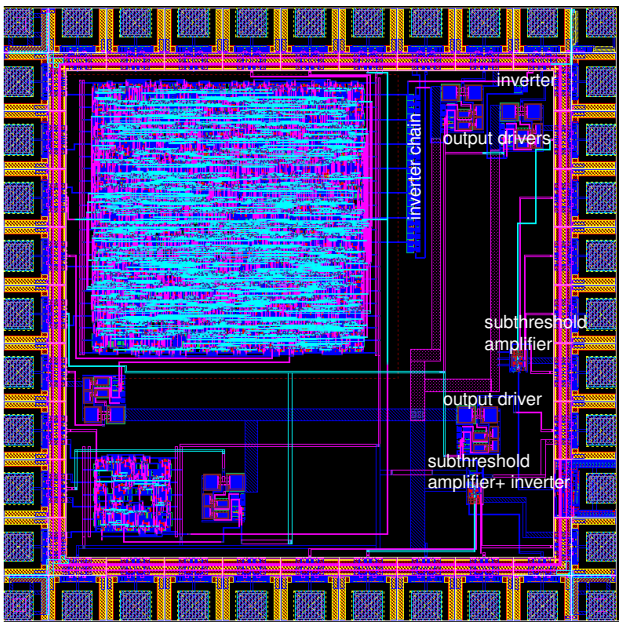


Figure 16: Die photo of the AMI 0.5 μm chip.

simulating and modeling complex phenomena and for selecting random samples from larger data sets. In this dissertation, RNGs are critical for the realization of probabilistic algorithms.

The choice of the RNG for a specific application depends on the requirements specific to the given application. If the ability to regenerate the random sequence is of crucial significance, or the randomness requirements are not very stringent, or the hardware generation costs are unjustified, then pseudo-random number generators (PRNGs) should be used. PRNGs are algorithms implemented on finite-state machines and are capable of generating sequences of numbers which appear random-like from many aspects. Though they are necessarily periodic, their periods are very long, they pass many statistical tests and can be easily implemented with simple and fast software routines. However, when ultimate security is required one must turn to the only cipher which is theoretically unbreakable [187]. This cipher requires a truly random sequence, and PRNGs are inappropriate for such a purpose. Thus, in all these cases where PRNGs are not suitable and unpredictability is a more important requirement than repeatability, one must turn to generators of truly random numbers. In this dissertation, in the context of RNGs, our primary purpose is to realize a high-quality RNG, which can produce the random numbers required by the probabilistic algorithms as

energy efficiently as possible. Due to the high quality requirement, we resort to true random number generators.

It is widely accepted that the core of any RNG must be an intrinsically random physical process. Hence, the proposals and implementations of RNGs range from tossing a coin, throwing a dice [119], drawing from a urn, drawing from a deck of cards and spinning a roulette to measuring radioactive decay from a radioactive source [222, 89], using chaotic maps [199, 200, 86], sampling a stable high-frequency oscillator with an unstable low-frequency clock [50, 203], and measuring thermal noise from a resistor and shot noise from a Zener diode or a vacuum tube [222, 141, 75, 164]. Among these true random generators, due to its implementability in CMOS and its simplicity, in this dissertation research, we preferred to implement a thermal noise based random number generator.

4.1.1.1 Design of the Thermal Noise Based RNG

In this section, we will describe the design of the thermal noise based RNG in a 0.25μm technology. This thermal based random number generator, also shown in Figure 17 is composed of a resistor, whose thermal noise is amplified, a two-stage common source amplifier, and a CMOS inverter digitizing the output of the amplifier. We refer to this structure as random event source integrated noise amplifier (RESINA). To minimize the energy consumption of this structure, the amplifier is designed such that the DC operating points of the transistors are in the subthreshold region of operation. In our design, the resistor (denoted as R in Figure 17) is connected externally. In our experiments, which will be described later in Section 4.2, initially, we used a resistance value of 2 $G\Omega$. Due to the requirements enforced by the probabilistic applications that can utilize the random bits generated by our RNG, we target a bandwidth of 1MHz. Hence, from $V_t = \sqrt{4kTRBW}$, the rms value (V_t) of the noise that will be generated by this resistor at room temperature (300K) will be 5.8 mV. Here, k is the Boltzmann's constant (1.3807×10^{-23}), T denotes the temperature measured in Kelvin, R denotes the resistance, and BW represents the bandwidth. Given this rms value of the noise, our target gain is around 100 so that we would be able amplify the rms value of the noise up to a level that can affect the output of the inverter significantly. Here,

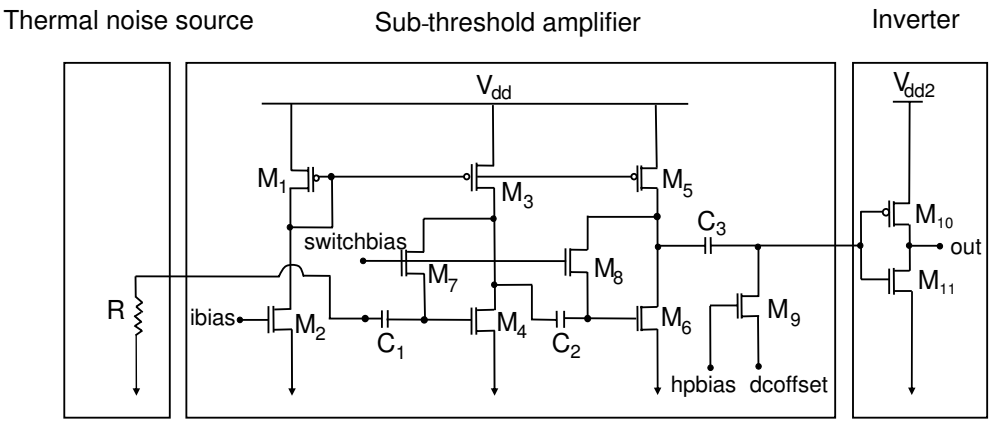


Figure 17: Thermal noise based random number generator.

we define the "gain" of the amplifier as the ratio of its output voltage magnitude to input voltage magnitude.

In the amplifier shown in Figure 17, NMOS transistors M_7 and M_8 help control the DC operating points of the common source stages by providing a feedback mechanism. The signal *switchbias* is used to control the DC bias voltages of M_7 and M_8 and vary the degree of the control of these transistors on DC operating points of the common source stages. For example, if switchbias is zero, then M_7 and M_8 are OFF and they have no effect on the common source stages.

The NMOS transistor M_9 accompanied by *dcoffset* and *hpbias* is used to control the DC offset value at the output.

C_1, C_2, and C_3 denote the decoupling capacitances. Furthermore, C_3 forms a capacitive divider with the input capacitance of the inverter. The transistors M_1 and M_2 constitute the current source of the amplifier. The signal denoted as *ibias* controls the current flowing on M_2, hence on M_1. This current is then mirrored to the common source stages. The magnitude of current on the common source stages would depend on the W/L ratio of the transistors M_3 and M_5 to that of M_1.

Since ibias determines the amount of current on the current source stage as well as the common source stages, it also affects the gain, bandwidth, and power consumption of the amplifier. For example, increasing ibias will increase the current leading to a higher bandwidth, but smaller gain.

The transistors M_4, M_6, M_3, and M_3 form the common source stages. The gain and bandwidth of these stages are determined by the current, as well as the W/L ratio of these transistors.

The gain of this amplifier can be estimated by

$$Gain = g_{m4} \cdot (r_{o3} \| r_{o4}) \cdot g_{m6} \, (r_{o5} \| r_{o6}) \, \frac{C_3}{C_3 + C_L} \tag{70}$$

where g_{m4} and g_{m6} denote the transconductance of the transistors M_4 and M_6, respectively. The output resistances of the transistors M_3, M_4, M_5, and M_6 are denoted by r_{o3}, r_{o4}, r_{o5}, and r_{o6}. The fraction $\frac{C_3}{C_3+C_L}$ is due to the capacitive divider formed by C_3 and C_L. Here C_L is the capacitive load for the amplifier and includes the drain capacitance of M_9 and the gate capacitances of M_{10} and M_{11}. The transconductance g_m of a transistor is expressed as

$$g_m = \left. \frac{\partial I_D}{\partial V_{GS}} \right|_{V_{DS},const} \tag{71}$$

where I_D is the channel current of the transistor, V_{GS} is the gate to source voltage and V_{DS} is the drain to source voltage. We note these are DC currents and voltages. Hence, g_m represents the sensitivity of the transistor current to a change in its input voltage. The conductance in the subthreshold region can be modeled as [212]

$$g_m = \frac{I_{DS}}{n \Phi_t} \tag{72}$$

where Φ_t is the thermal voltage (kT/q). n is expressed as

$$n = 1 + \frac{\gamma}{2\sqrt{2\phi_f + V_{SB}}} \tag{73}$$

where ϕ_f is the intrinsic Fermi voltage [212], γ is the body effect coefficient [212], and V_{SB} is the source to substrate voltage.

On the other hand, the output resistance of a transistor is expressed as

$$r_o = \left. \frac{\partial V_{DS}}{I_D} \right|_{V_{GS},const} \tag{74}$$

and reflects the effect of channel-length modulation on the transistor behavior. In the subthreshold region, the output resistance can be modeled as [212]

$$r_o = \frac{1}{\lambda_W I_{DS}} \tag{75}$$

where λ_W is the channel length modulation factor.

The bandwidth of our amplifier can be estimated by

$$BW = \frac{1}{(r_{o5} \| r_{o6}) \cdot \left(C_{d5} + C_{d6} + \frac{C_3 \cdot C_L}{C_3 + C_L} \right)} \tag{76}$$

The power consumption of the amplifier can be estimated using

$$Power = V_{dd} \cdot (I_{M_1} + I_{M_3} + I_{M_5}) \tag{77}$$

where I_{M_1}, I_{M_3}, and I_{M_5} denote the DC currents drawn by the transistors M_1, M_3, and M_5, respectively.

To ensure that the transistors in our amplifier are in subthreshold region, we chose the values of the bias voltages close to the threshold voltage (0.5 V) of this process. As a result, we started the amplifier design with the initial parameter set shown in Table 4.

Table 4: Initial parameter set for the amplifier.

$(W/L)_1 = (W/L)_3$	4.92 μm / 0.25 μm
$(W/L)_2 = (W/L)_4$	4.92 μm / 0.25 μm
$(W/L)_7 = (W/L)_8 = (W/L)_9$	3 μm / 0.25 μm
C_1	300.7 fF
C_2	156.5 fF
C_3	156.5 fF
V_{dd}	450 mV
switchbias	400 mV
hpbias	500 mV
dcoffset	500 mV

Starting with this initial parameter set, we investigated the effect of *ibias* and capacitances C_1, C_2, and C_3 on the gain, bandwidth, and the power consumption of the amplifier using circuit simulations. Circuit simulations are realized in Cadence Virtuoso®Analog Design Environment using the Spectre simulator and TSMC 0.25 μm transistor models available from MOSIS [133]. A load capacitance of 7 fF is used in these simulations. Table 5 below shows the variation of the gain, bandwidth, and power of the amplifier as ibias is varied. For these simulations, the amplifier uses the initial parameter set.

It is seen from Table 5 that increasing ibias increases the current flow in the circuit which leads to a higher power consumption. Therefore, even though increasing ibias gives a higher

Table 5: The effect of ibias on the gain, bandwidth and power of the amplifier.

ibias (mV)	Gain	Bandwidth (MHz)	Current (nA)	Power (nW)
390	99.11	1.51	724.69	471.05
385	118.62	1.32	655.66	426.18
380	127.71	1.17	575.57	374.12
375	135.38	1.06	518.69	337.15
370	139.86	0.97	466.28	303.08

bandwidth, we should consider the trade-off between the bandwidth, power consumption and gain, that is, higher bandwidth leads to higher power consumption and lower gain. Hence, a moderate value of ibias such as 380 mV is preferable.

Among the decoupling capacitances C_1, C_2 and C_3, we observed that C_3 has the most significant effect on the gain of the circuit. For example, for the amplifier with the parameters given in Table 5 and ibias of 380 mV, when C_3 is increased from 156.5 fF to 500 fF, the gain of the circuit increases from 127.61 to 141.8. This is due to the capacitor divider formed by C_3 and C_L. Hence, C_L also affects the gain of the circuit. For example, for the amplifier with the parameters given in Table 4 and ibias of 380 mV, when C_L is increased from 7 fF to 60 fF, gain decreases from 127.61 to 104.79. Thus, it is preferable to choose a higher value of C_3, such as 500 fF, and a lower value of C_L. Since in the thermal noise based RNG, which is the topic of this section, amplifier output is connected to an inverter, we could minimize C_L by choosing the minimum size inverter.

We have not observed any significant effect of transistor sizes on the gain and bandwidth of the amplifier.

4.1.1.2 Post-fabrication Results for the Subthreshold Amplifier

Following the schematic level simulations, the amplifier design was extracted using the TSMC 0.25 μm SCN5M_DEEP design rules available from MOSIS. Figures 18 and 19 show the transient and AC simulation results for the extracted circuit. Here, ibias is 380 mV, C_3 is 500 fF, and C_L is the capacitive loading due to a minimum size inverter from the VTVT standard cell library [226]. The other parameters are as shown in Table 4. As seen from Figure 18, amplifier gain is around 150. We could also see from figure that the DC

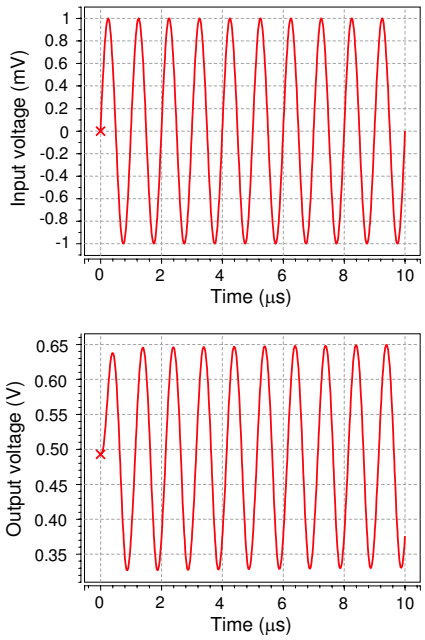


Figure 18: Transient response of the amplifier: Input and output voltage waveforms.

operating point of the amplifier takes a few seconds to stabilize. DC offset at the output of the amplifier is nearly $\sim$ 500 mV since the magnitude of the dcoffset signal is 500 mV. From Figure 19, it is seen that the flat-band gain of the amplifier is 44.28 dB, corresponding to a ratio of 163.78. The frequency values at which the gain decreases from its flat-band value by 3 dB are 1.819 MHz and 102.45 Hz. Hence, the bandwidth of the amplifier is $\sim$ 1.8 MHz. Hence, this amplifier design is well within our requirements of gain of 100, and bandwidth of 1MHz. Furthermore, the power consumption of this amplifier is measured as 371.78 nW.

4.1.1.3 Post-fabrication Results for Thermal Noise Based RNG

In this section, we describe the post-fabrication results for our thermal noise based RNG. This RNG includes the extracted amplifier that we described in the former section. Since it is not possible to do a transient simulation of the thermal noise of a resistor, noise is generated as described before in Section 3.3. The CMOS inverter component of the RNG is a minimum sized inverter from the VTVT standard cell library [226].

Figure 20 depicts the voltage waveforms at the output of the amplifier and RESINA. Here, the rms value of the input noise is 0.5 mV. The resulting noise at the output of the amplifier has a mean value of 0.55 V and rms value of 64.59 mV. The mean value of the

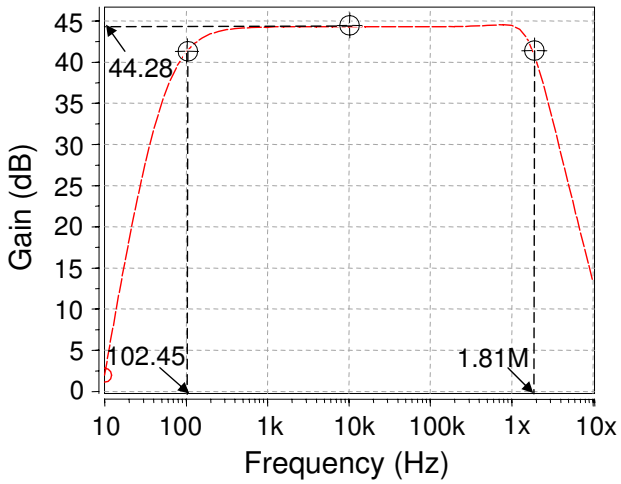


Figure 19: Amplifier gain versus frequency.

output is different from 0 since the amplifier has a DC offset at the output. If we decrease the value of dcoffset to 0, then DC offset at the output of the amplifier will be 0. Supply voltage of the inverter is 1 V, therefore we observe from Figure 20 that the pulses at the output of RESINA are between 0 and 1V. We note that there is a DC offset of approximately 490 mV at the input of the inverter of RESINA. This DC offset value is greater than V_m of this inverter, which is measured to be 0.478 V through simulations. As a result, we expect the output of RESINA to be a binary 0. Therefore, in Figure 20 an error occurs whenever there is a transition to binary 1. To reiterate, the inverter of RESINA is coupled to a Gaussian distributed noise at its input wherein noise has a mean value of $\mu = 0.55$ V and rms value of $\sigma = 64.59$ mV. For such an inverter, the probability of output being correct (output being binary 0) can be found from

$$p = \frac{1}{2}\left[1 - erf\left(\frac{V_M - \mu}{\sqrt{2}\sigma}\right)\right] \tag{78}$$

From (78), we find the probability of correctness to be 0.867, which is very close to simulated result, 0.877.

Since the validity of (78) is based on the validity of the Gaussian distribution approximation for the noise at the input of the inverter of RESINA, we also investigated the distribution of the amplified noise. In Figure 21, we show the probability density function of the noisy voltage at the output of the amplifier, which is denoted as V_{amp} in the figure.

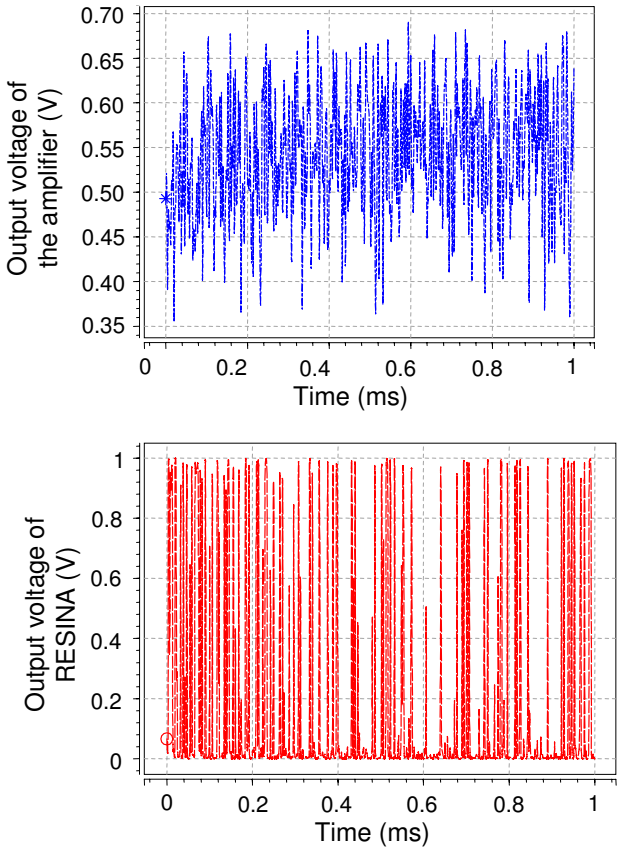


Figure 20: Voltage waveforms at the output of the amplifier and RESINA in case when rms value of input noise is 0.5 mV.

As seen from the figure, the distribution of the noise approximates a Gaussian distribution with mean value of 0.55 V and rms value of 64.59 mV. When V_{amp} is in the range 0.47 V to 0.49 V, the amplitude distribution differs from the ideal Gaussian distribution the most since DC offset value is in this range. We also note that when the output voltage of the amplifier is greater than the supply voltage value of the amplifier (which is 650 mV) the output voltage of the amplifier becomes distorted, that is the the output voltage is no more an exact amplified version of the input voltage, and the power of the output voltage is smaller than $gain \times |V_{in}|^2$. As a result, the probability density of the output voltage value decreases faster than the probability density of an ideal Gaussian voltage when output voltage is smaller 0.65 V.

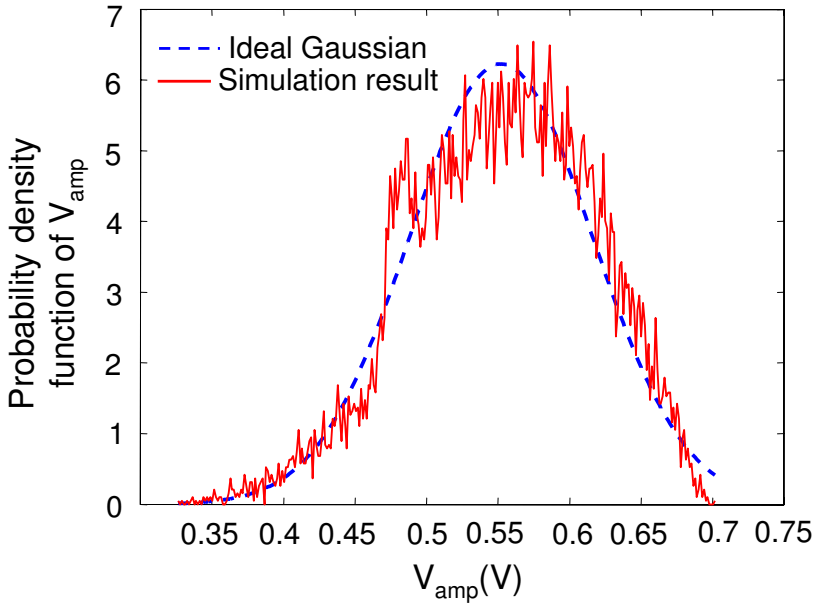


Figure 21: The distribution of the noise at the output of the amplifier.

4.2 Measurement Framework

There are four different types of measurements that were performed on our chips. These are

- Functionality testing of each component
- Current measurement of each component
- Measurement of the quality of the random bits generated by the thermal noise based RNG
- The *E-p* relationship characterization of the inverter

4.2.1 Functionality Testing of the Subthreshold Amplifier

Functionality testing of the subthreshold amplifier is done using an oscilloscope (HP 54645D) and a function/arbitrary waveform generator (HP 33120A). A sinusoidal input is applied at the input of the amplifier. The expected gain of the amplifier is ~ 117 and its expected bandwidth is ~ 900 kHz. We note that these values are different from the values we listed in Section 4.1.1.3. This is due to the difference in the load capacitance values between the

case of the simulations performed in Section 4.1.1.3, and the case of a standalone amplifier. The output of the standalone amplifier is driven by an output buffer, which has an input capacitance value of 150 fF. In contrast, in the case of the simulations we performed in Section 4.1.1.3, we considered the case when the capacitive load of the amplifier is due to a minimum size inverter.

Varying the dcoffset signal changes the DC offset at the output of the amplifier. Changing the bias signal changes the current drawn by the amplifier, hence it changes the bandwidth of the amplifier. Based on the layout based HSPICE simulations that we summarized in the former section, a dcoffset value of 500 mV, a switchbias value of 400 mV, an hpbias value of 500 mV, a bias value of 380 mV and a supply voltage value of 450 mV are used.

Among the instruments that are used for this experiment, HP 54645D is a 100 MHz scope with 200 MSample/s. It has two analog channels and 1 MB of memory per scope channel. Its maximum vertical sensitivity is 5 V/div and minimum sensitivity is 1 mV/div. It has $\pm 1.5\%$ DC gain accuracy. On the other hand, HP 33120A can generate 15 MHz sine and square wave outputs. It can also generate 100 kHz square and ramp outputs, as well as white noise with 10 MHz bandwidth. The frequency resolution of this instrument is 10 μHz and its accuracy is 20 ppm in one year. The output amplitude can be in the range 50 mV (peak-to-peak) to 10 V(peak-to-peak). The accuracy of its amplitude is $\pm 1\%$ of the specified output at 1 kHz.

We also designed a printed circuit board (PCB) to test the functionality of the amplifier and the thermal noise based RNG. We used a four-layer PCB board which has top and bottom copper layers with all holes plated through, and ground and power planes that are sandwiched inside. This provides good grounding and reduces the effect of power supply noise and ground bounce noise. Figure 22 shows the schematic of the PCB. As shown in the figure, the board includes adjustable regulators which help produce the bias and supply voltages required by the amplifiers, OPAMPs which are used to buffer supply and bias voltages, and decoupling capacitors to decouple the power supply lines.

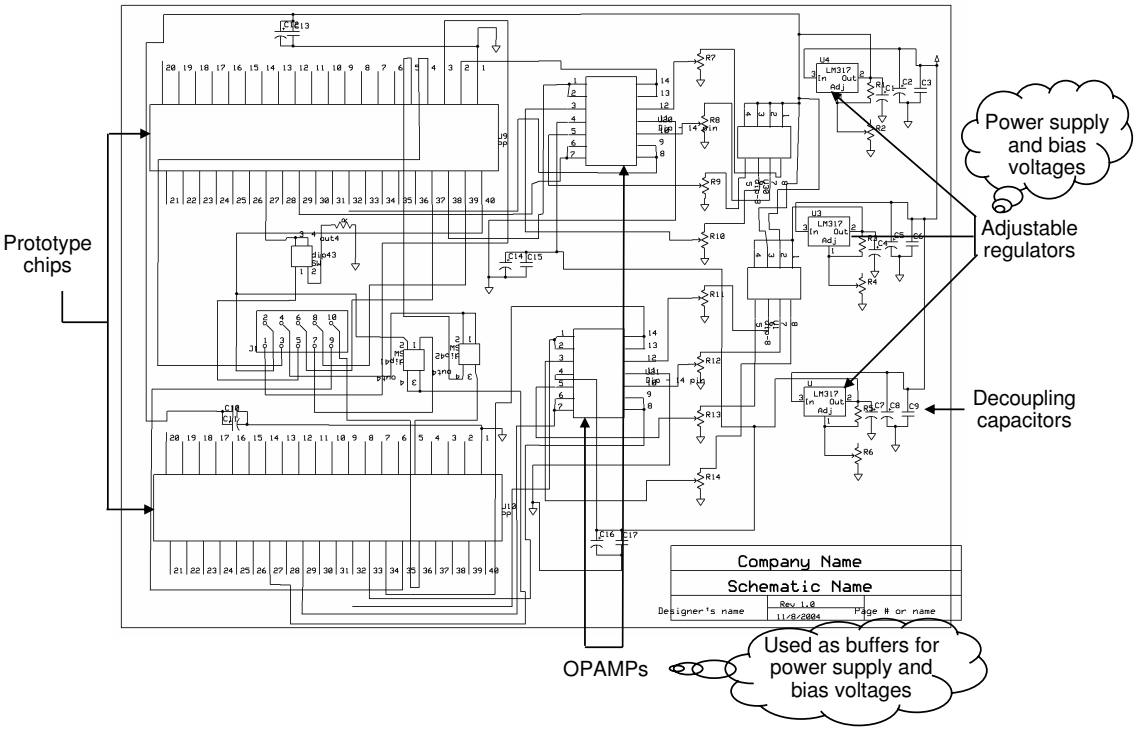


Figure 22: Schematic of the PCB designed to test the amplifier and RESINA.

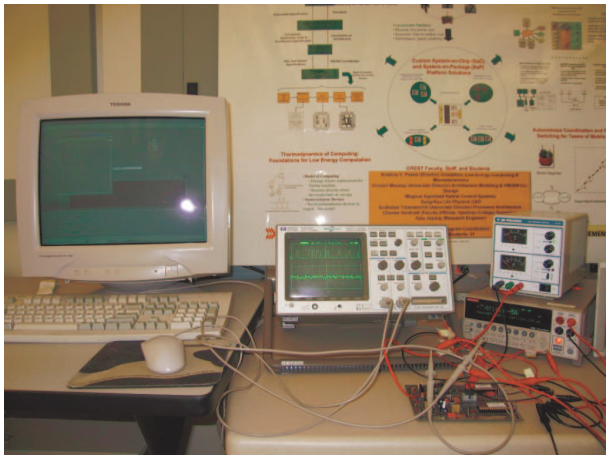


Figure 23: Measurement setup for functionality testing of RESINA.

4.2.2 Functionality Testing of the Thermal Noise Based RNG

In this experiment, functionality of RESINA is tested. Here, the resistor is connected externally. Initially, a big resistor (2 $G\Omega$) was used, however due to bandwidth considerations that will be clarified later in Section 4.3, a 1 MΩ resistor was used in later experiments. In order to reduce the external interference, the resistor is shielded with an aluminum box. Output voltage of RESINA is captured by the oscilloscope and downloaded to a host computer so that these random bits can later be tested in terms of their quality. The interface used between the oscilloscope and the host computer is General Purpose Interface Bus (GPIB), which is also referred to as IEEE-488. The interface software is Labview. Figure 23 shows a photo of the measurement setup.

4.2.3 Measurement of the Average Current Consumed by Each Component

Keithley 2400 sourcemeter is used to measure the current consumed by each component. This sourcemeter has a maximum resolution of 10 pA with an accuracy of 0.029% + 300 pA while measuring current values smaller than 1 μA. This instrument can take current measurements in the range from 10 pA to 1.055 A.

Since our aim is to measure the switching current drawn by each component, when one component is under test other components are disabled, that is, they do not switch. Below, we describe how each component is disabled.

- Inverter and inverter chain are disabled by keeping their input voltages constant.
- The subthreshold amplifiers are disabled by keeping their bias signals at 0.

Figure 24 shows a photo of the PCB board and Keithley 2400 sourcemeter while the sourcemeter is being used to measure the current drawn by a PCMOS inverter. As it is seen from the figure, there are many cables that has to be connected to the external power supplies, since the amplifiers require many bias voltages.

4.2.4 Measurement of the Quality of the Random Bits

The measurement of the quality of the random bits necessitates the collection of a large number of output bits—for some tests $20,000,000$ data points are required. We use HP

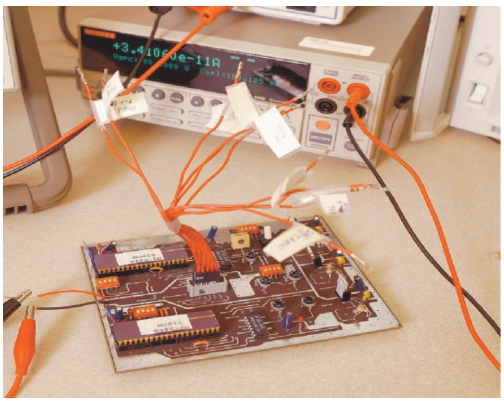


Figure 24: Keithley sourcemeter while being used to measure the current drawn by a PCMOS inverter.

54645D to collect the output data of the thermal noise based RNG. Then, the randomness quality tests [174] are run on the collected data. Since some of these tests require bit numbers as high as $20,000,000$, and since the oscilloscope memory is limited to 1 MB, we need to take many consecutive measurements.

4.2.5 Characterization of the Relationship Between Energy and Probability of PCMOS Switches Using Physical Measurements

To characterize the energy and probability relationship of the inverter, we need to measure the current drawn by the inverter and its probability of correctness.In Chapter 3, we identified three different ways that the noise is coupled to the inverter: output coupling, input coupling, and power supply line coupling. In hardware measurements, noise is coupled at the input of the probabilistic inverter, since it would be very complicated to couple the noise to the output and there would be interference of the power supply noise if the noise was coupled to the power supply line. Hence, to find the probability of correctness of a PCMOS inverter, Gaussian noise is applied at the input of the inverter and the resulting output is captured on the HP 54645D. Agilent 33120A is used to generate the Gaussian noise.

4.3 Measurement Results

In this section, we will summarize our physical measurement results starting from the results regarding the thermal noise based RNG and the subthreshold amplifier. Following this, we will describe the measurement results for the energy-probability characterization of PCMOS

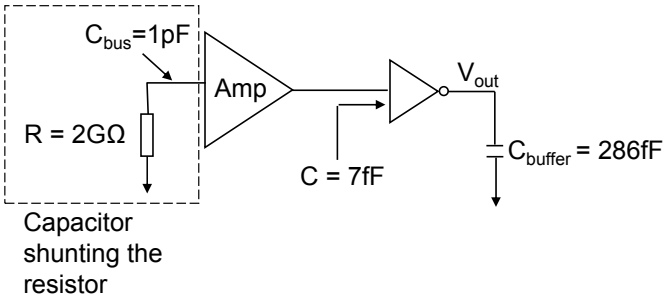


Figure 25: The capacitor resulting from the bus capacitance is shunting the resistor.

switches. Finally, we will present the results of randomness quality tests for the random bits measured from our thermal noise based RNG and we will compare the quality of our random bits to the quality of random bits produced by a PRNG implemented in CMOS.

4.3.1 Measurement Results for the Thermal Noise Based RNG

During the measurements on the thermal noise based RNG, we found that the maximum frequency component of the noise is much smaller than 1 MHz. This was resulting from the fact that our resistor was shunted by the capacitor. As shown in Figure 25, our experimental setup includes a 2 GΩ resistor in series with an amplifier. Since the resistor is connected externally, it is in series with a large bus capacitance, which we estimated as 1 pF. Due to this capacitance, the mean-square value of the thermal noise of the resistor is now found from [134]

$$v_t^2 = \int_0^\infty \frac{4kTR}{1 + (2\pi f RC)^2} df \tag{79}$$

where C and R denote the capacitance and resistance values, respectively. f and T denote the frequency and temperature of the circuit, while k is the Boltzmann's constant. After computing the integral in (79), the expression for v_t^2 is further simplified to

$$v_t^2 = \frac{kT}{C} \tag{80}$$

which indicates that the mean-square value of the thermal noise of a resistor shunted by a capacitance depends only on the value of the shunt capacitance C. We note that, the maximum frequency component of the noise is determined by the time constant, RC, of this resistance-capacitance network.

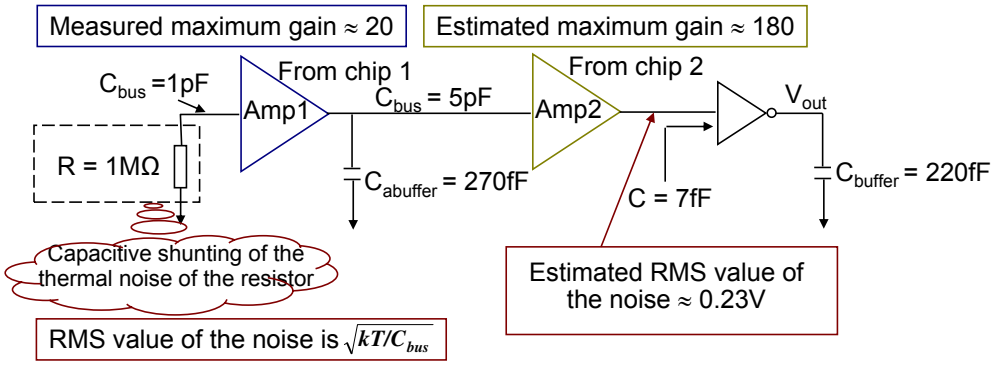


Figure 26: RESINA design with two cascaded amplifiers.

Reconsidering our design with the resistor being shunted by the capacitor, we decided to select a resistor with a resistance of 1 MΩ. However, now the rms value of the noise is $\sqrt{kT/C}$, which is 6.44×10^{-5} and when this rms value of noise is amplified by a factor 150, the resulting noise is still very small. Therefore, we cascaded two amplifiers to increase the gain, and hence to increase the amplitude of the resulting noise. The design with the cascaded amplifiers is shown in Figure 26.

For this cascaded amplifier design, we measured the gain of the first stage to be 20. We note that we could change the gain of the second stage by changing the value of the ibias signal. For example, when the value of the ibias signal is decreased from 390 mV to 350 mV, gain of second stage increases from 150 to 180. Furthermore, if we do not prefer to keep the DC offset value at 0, then because of the DC offset at the input of the inverter the inverter would be subject incorrect switchings even when the rms value of noise is as small as 65 mV as we showed in Section 4.1.1.3. In this case, the gain of the second stage could be smaller than 150.

Table 6: Bias voltages for the two cascaded amplifiers on which physical measurements are taken.

	Amp1	Amp2
Supply voltage	500 mV	650 mV
bias	390 mV	380 mV
switchbias	420 mV	420 mV
hpbias	520 mV	0
dcoffset	500 mV	0

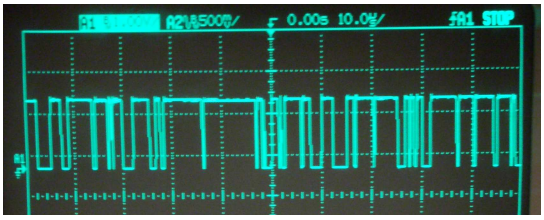


Figure 27: Output of RESINA measured on an oscilloscope when the amplifier parameters are as shown in Table 6.

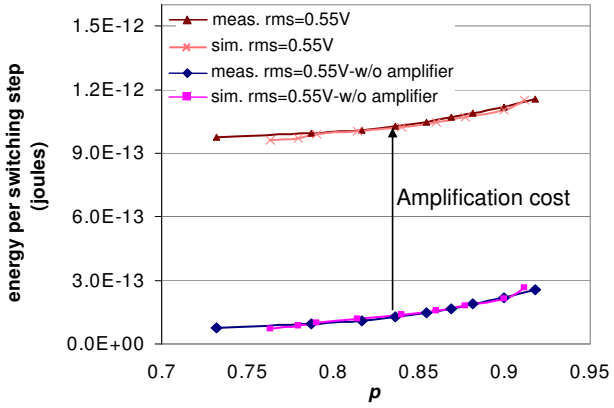


Figure 28: Energy-probability relation for RESINA designed using a 0.5 μm technology.

Figure 27 shows a photo of the output voltage of RESINA measured on oscilloscope. For this measurement, the bias voltages for the two cascaded amplifiers are listed in Table 6. The probability of correctness in this case is measured to be 0.89. In this experiment, the supply voltage of the inverter is 1 V.

By varying the supply voltage of the inverter, we found the energy-probability relationship for the thermal noise based RNG. In Figure 28, we depict the energy-probability relationship for a design using a 0.5 μm technology for the case when the rms value of the resulting noise is estimated to be 0.55 V. As seen from the figure, the energy cost of amplification is very significant. For the RESINA design in the 0.5 μm technology, the amplifier consumes 900 fJ per switching of RESINA. For a smaller feature size of 0.25 μm technology, we measured the amplifier energy consumption to be 400 fJ.

4.3.2 Quality of the Random Bits Produced by the Thermal Noise Based RNG

The *quality* of randomization also plays an important role in many probabilistic algorithms. Among these algorithms that are sensitive to quality of randomness, Monte Carlo simulations, for example, have been shown to yield incorrect results when poor quality pseudo-random number generators are used as the source of random bits [51]. In addition, the strength of encryption schemes like hyper-encryption [45] can be severely compromised unless true-random (as opposed to pseudo-random) sequences are provided. For such applications whose correctness, and hence utility, depend on the quality of random bits, it is important to compare the quality of randomization afforded by PCMOS and conventional CMOS based implementations.

In Figure 29, we show the statistical tests from the NIST suite [174] applied to compare and evaluate the quality of random bit sequences generated by a PCMOS inverter (switch) and those generated by a CMOS-based pseudo-random number generator; for both cases, p is considered to be 0.5. The random sequences in the case of PCMOS have been produced from the actual chip measurements of a PCMOS inverter from 0.25 μm TSMC prototype and those of CMOS from HSPICE simulations of the hardware implementation of the Park-Miller [160] random number generator.

Among these tests and to highlight a few, the *frequency* tests evaluate the frequency of 0s and 1s—whether the fraction of 1s to 0s is close to 0.5. The *runs* test determine contiguous sequence of 1s in a block. The *rank* test checks the linear dependence, while the *FFT* and *approximate entropy* tests detect periodicity and frequency of overlapping patterns. The *template* matching tests detect repetitions of non-periodic patterns and the *universal statistical* test as well as the *lempel-ziv* test detect whether the random sequence can be compressed. The tests are run on random bit sequences of length 20,000,000. In evaluating the test results, we use the same testing strategy and criteria as recommended by the NIST suite. Specifically, the test results shown in parenthesis in the table are compared against a threshold (which is 0.93) used to determine whether the sequence passes or fails a test. The result indicates the proportion of subsequences (tested through iterations) that pass from the random sequence being tested. As seen from the figure, PCMOS passes 11

Test	PCMOS	CMOS
Frequency	PASS (0.98)	FAIL (0.84)
Block-frequency	PASS (1.00)	PASS (0.98)
Cumulative sum	PASS (0.98)	FAIL (0.86)
Runs	PASS (0.98)	PASS (0.96)
FFT	PASS (1.00)	PASS (1.00)
Approximate entropy	PASS (0.98)	FAIL (0.92)
Long-run	PASS (1.00)	PASS (1.00)
Rank	PASS (1.00)	FAIL (0.00)
Non-overlapping template	PASS (0.9375)	PASS (0.9375)
Overlapping template	FAIL (0.8889)	FAIL (0.00)
Lempel-Ziv	FAIL (0.8125)	FAIL (0.0625)
Linear complexity	PASS (1.00)	PASS (1.00)
Universal Statistical	FAIL (0.725)	FAIL (0.8889)
Serial	PASS (1.00)	PASS (1.00)

(result > 0.93) → PASS
(result < 0.93) → FAIL

Figure 29: Comparison of quality of randomization for PRNG and PCMOS.

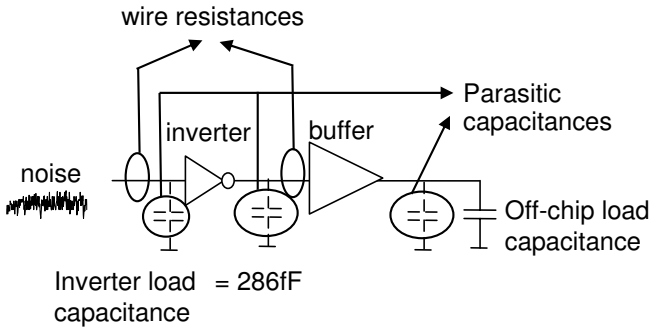


Figure 30: Probabilistic inverter, its output buffer and parasitic elements.

tests out of 14 whereas CMOS passes only 7 of these tests.

4.3.3 Measurement Results Characterizing the Energy-Probability Relationship of an Inverter

This section summarizes the physical measurement results for a probabilistic inverter coupled to noise at its input. These results involve current measurements as well as the results of probability measurements.

Below, we also show a comparison of the measurement results to analytical results. To compare these two, we first model the measurement circuitry as shown in Figure 30. As seen from the figure, the output of the inverter is connected to a buffer that can drive large off-chip capacitances. While measuring the current driven by the inverter, a large off-chip

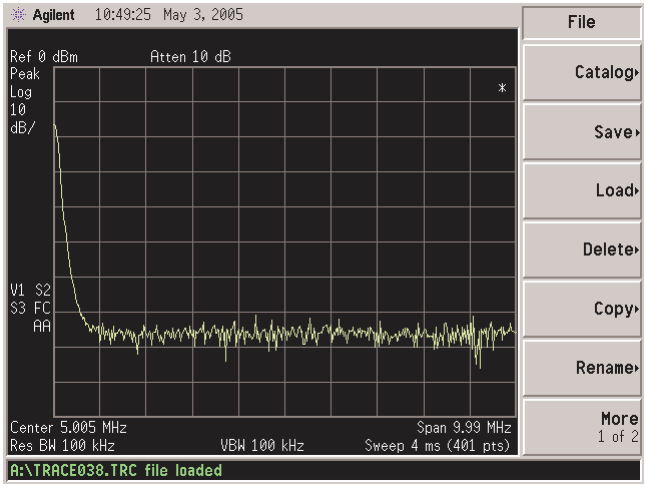


Figure 31: Power spectral density of the output voltage of a probabilistic inverter coupled to noise at its input.

capacitance is connected to the output of the buffer so that buffer does not switch and contribute to the current consumption. On the other hand, for the probability measurements, we do not connect any external capacitors, and we estimate the load capacitance seen by the output of the buffer as 10 pF. The buffer is designed to work at a frequency of 10 MHz for this value of load capacitance. In Figure 31, we show the power spectrum of the noise at the output of the inverter. We note that, in this case, the only input applied to the inverter is the Gaussian noise generated by the function/arbitrary waveform generator. As seen from the figure, for a span of almost 10 MHz, the power spectral density of the output is nearly constant. Hence, the maximum frequency component of the noise after being filtered by the RC networks due to parasitic capacitances and resistances is larger than 10 MHz.

Figure 32 shows the measurement and analytical results for the energy-probability relationship for 0.25 μm inverter at a range of rms values of noise. As seen from the figure, there is a gap between the analytical and simulation results. We found that this gap is primarily caused by the difference between the analytical and measured energy values. This difference is due to the fact that our analytical model does not consider the short-circuit energy consumption. Due to the parasitic resistance and capacitances, the rise and fall times of the input signal of the probabilistic inverter is much higher than its switching time leading to a significant short-circuit energy consumption. Therefore, we improved our analytical model

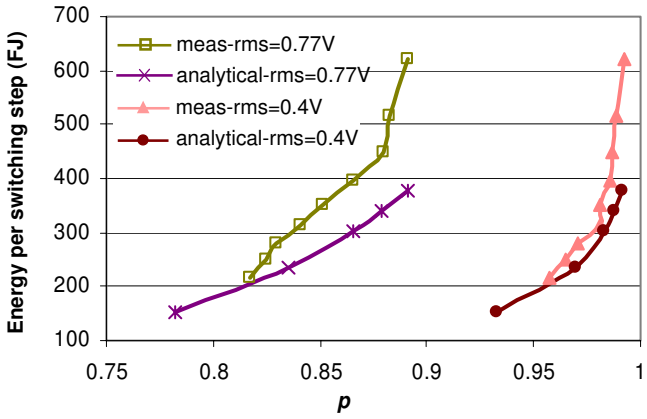


Figure 32: Measurement results compared to the analytical results for the energy-probability relationship for a 0.25 μm inverter.

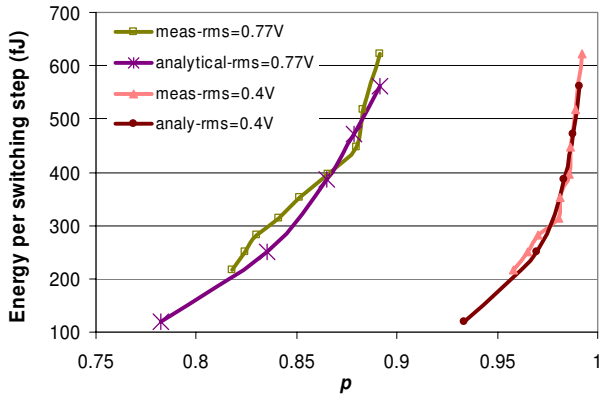


Figure 33: Validation of the improved analytical model for the energy-probability relationship of 0.25 μm inverter by comparison against measurement results.

by developing a short-circuit energy model. Following this chapter, in Chapter 5, we will describe the short-circuit energy model in detail. In Figure 33, for the energy-probability relationship of the same 0.25 μm inverter, we compare the physical measurement results with the results found using the improved analytical model that includes the short-circuit energy. As seen from the figure, the results produced by the improved analytical model match the physical measurement results closely.

4.4 Conclusions

In this chapter, we summarized the physical measurements performed to validate the CMOS behavior. Furthermore, we described the design of thermal noise based RNG, and the measurements for testing its functionality and probabilistic behavior. The measurement

results showed that in the thermal noise based RNG, the large bus capacitance shunts the resistor leading to a reduction in the maximum rate of random bits produced by our RNG. As a result, the resistance value used in RNG is decreased and the design is changed to include two amplifiers to compensate for the gain loss due to the decrease in the resistance value. The measurement results also showed that the energy cost of amplification is very significant, but it decreases as the technology progresses.

Measurements were also used to derive the energy-probability characteristics of a probabilistic switch. When the measurement results were compared to analytical results, a significant gap was observed between the two. This gap was due to short-circuit energy consumption, which was subsequently included in the analytical model leading to a strong match between the measurement results and the results produced by the improved analytical model.

CHAPTER V

AN IMPROVED ENERGY MODEL FOR THE PCMOS INVERTER

In Section 4.3, it was seen that there may be a significant gap between the measurement and analytical results when the analytical model does not include the short-circuit energy consumption. Hence, this chapter is devoted to developing an analytical model for the short-circuit energy consumed by a PCMOS inverter.

We have developed a short-circuit energy model following Bisdounis and Koufopavlou [20]. In our model, we explicitly use the alpha-power (α-power) law MOSFET model of Nose and Sakurai [151].

In this chapter, we will first explain the reasons for short-circuit energy consumption and describe the prior work on this subject. Following this, we will detail our short-circuit energy model starting with the derivation of the short-circuit current. This will be followed by a validation of the model using circuit simulations. Finally, we will conclude this chapter.

5.1 Background

The short-circuit energy dissipation results due to a direct path current flowing from the power supply to the ground during the switching of a static CMOS gate. Short-circuit energy constitutes 10-20% of the total energy dissipation of a static CMOS gate [5].

The first closed form expression modeling the short-circuit energy dissipation in a CMOS inverter was developed by Veendrick [220], where zero-load capacitance is assumed. The model results in pessimistic results because of the zero-load capacitance assumption. In addition, the model is based on the Shichman and Hodges square law MOSFET model [191], which ignores the short-channel effects of the submicron devices.

In [69], a more realistic short-circuit energy dissipation model was proposed. This model includes the effect of the output load capacitance. However, short-channel effects are

ignored in the derivation of the model. In [221], the model presented in [69] is improved by including the velocity saturation effects through use of alpha-power (α-power) law MOS model [183]. However, the contribution of the PMOS (NMOS) currents in falling (rising) output is neglected in the derivation of the model. In addition, Miller effect of the gate-to-drain capacitance is not included. Furthermore, it is assumed that the load transistor operates only in the saturation region during the time interval in which the short-circuit current flows. In [230], another short-circuit energy dissipation model based on Shichman and Hodges MOSFET current model was proposed. This model considers the effect of the PMOS (NMOS) current on the short-circuit current in case of the falling (rising) edge of the output through use of two technology dependent empirical parameters. Miller effect of the gate-to-drain capacitance is also included in the model. To include this effect, technology dependent empirical parameters are used.

In [72], short-circuit current waveform was approximated with a piecewise linear function of the time to estimate the short-circuit energy dissipation. In this model, energy dissipation of the reverse current due to the gate-to-drain capacitance is subtracted from the short-circuit energy dissipation. However, this reverse current is provided from the inverter input, but not from the power supply of the gate, hence this energy component can not be included in the short-circuit energy dissipation.

In [20], a short-circuit energy model was developed using the α-power law MOS model [183]. The model takes into account the current through both transistors. The influence of the gate-to-drain capacitances of both transistors and the gate-to-source capacitance of the short circuiting transistor are included in the derivation of the model. However, the α-power law MOS model does not very well capture the short-channel effects.

Nose and Sakurai [151] derived a closed-form expression for modeling the short-circuit energy dissipation of a CMOS inverter. They used an improved version of the alpha-power law MOSFET model [182] in their derivation, which is more accurate especially in the triode region than the original alpha-power law model [183]. Their model includes the short-channel effects. However, it does not include the effect of the gate-drain and gate-source capacitance of the transistors.

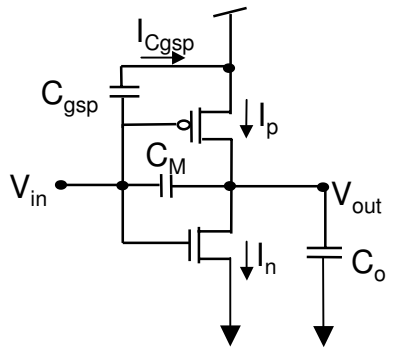


Figure 34: An overview of the important parameters that affect the short-circuit energy dissipation of a CMOS inverter.

In this work, we improve the model in [20] by using the alpha power law MOS model of [151].

5.2 Modeling the Short-Circuit Energy Dissipation of a PCMOS Inverter

In this section, we derive the short-circuit energy dissipation in a CMOS inverter (shown in Figure 34) for the rising input. The derivation for the falling input could be done similarly.

Figure 34 shows a CMOS inverter and the capacitance and current values that are considered in our model. Here, the output capacitance C_o includes the drain junction capacitances of the two transistors of the inverter, the gate capacitances of the fan-out gates, and the interconnect capacitance. C_M is the Miller capacitance and is equal to the sum of the gate to drain capacitance of both transistors. C_{gsp} is the gate to source capacitance of the PMOS transistor. I_p and I_n are the drain to source currents of the PMOS and NMOS transistors respectively.

The input, V_{in}, is modeled as $V_{in} = V_{DD}.(t/\tau)$ for $0 \leq t \leq \tau$, where τ is the input rise time. The differential equation in (81) describes the discharge of the output capacitance C_o.

$$C_o = \frac{dV_{out}}{dt} = C_{in}\left(\frac{dV_{in}}{dt} - \frac{dV_{out}}{dt}\right) + I_p + I_n \tag{81}$$

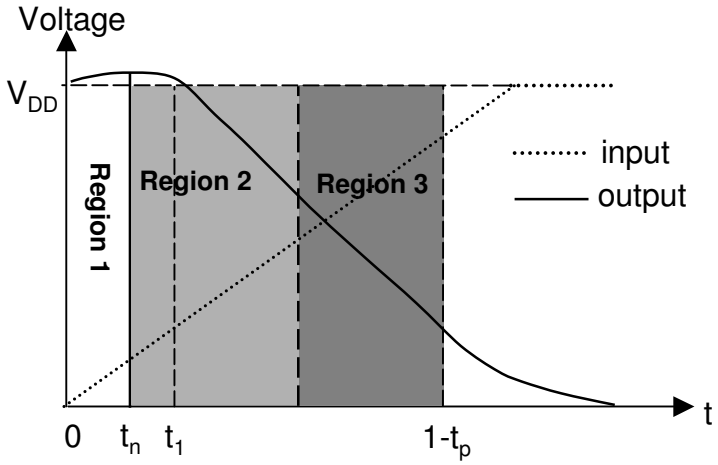


Figure 35: Operating regions of a CMOS inverter during a rising input.

5.2.1 Modeling the Short-Circuit Current of a CMOS Inverter

To compute the currents I_p and I_n, we use the α-power law MOSFET current model [151] also summarized below by (82) for a PMOS transistor. This model consists of four parameters: α, I_{DO}, V_{DO} and V_{TH}. α represents the velocity saturation index, which is an empirical parameter. I_{DO} is the drain current at $V_{GS} = V_{DS} = V_{DD}$ and V_{DO} is the drain to source saturation voltage at $V_{GS} = V_{DD}$. V_{TH} represents the threshold voltage of the α-power model and it is not the same as the physical threshold (V_{th}) of the transistor.

$$I_p = \begin{cases} 0, & |V_{GS}| < V_{THP} \quad \text{(cutoff region)} \\ I_{D_p} \cdot \left(2 - \frac{|V_{DS}|}{V_{D_p}}\right) \frac{|V_{DS}|}{V_{D_p}}, & |V_{DS}| < V_{D_p} \quad \text{(linear region)} \\ I_{D_p}, & |V_{DS}| > V_{D_p} \quad \text{(saturation region)} \end{cases} \tag{82}$$

where

$$\begin{aligned} I_{D_p} &= I_{DOP} \left(\frac{|V_{GS}| - V_{THP}}{V_{DD} - V_{THP}}\right)^{\alpha_p} \\ V_{D_p} &= V_{DOP} \left(\frac{|V_{GS}| - V_{THP}}{V_{DD} - V_{THP}}\right)^{\alpha_p/2} \end{aligned} \tag{83}$$

In modeling the short-circuit energy dissipation, we first derive analytical expressions for the output voltage of the inverter. Figure 35 shows the input and output voltage waveforms and the operating regions for the rising input.

In Figure 35, in region 1 ($0 \leq t \leq t_n$), NMOS transistor is OFF and PMOS transistor is in the linear region. Hence, in this region, I_n is approximated as 0. I_p is found from (82). By substituting the expression of I_p into (81), the differential equation can be solved for

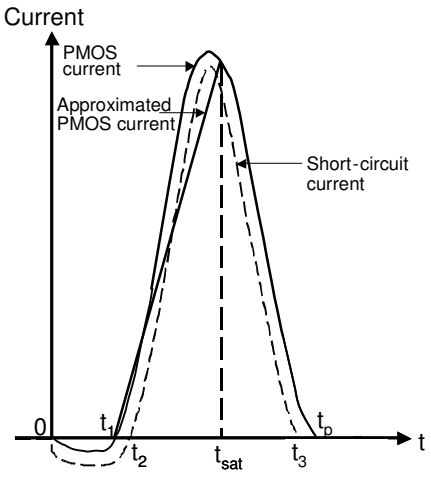


Figure 36: The PMOS and short-circuit current waveforms of a CMOS inverter during the rising edge of the input.

V_{out}. To simplify the problem, I_{D_p} and V_{D_p} are approximated to be constant, and equal to I_{D_pm} and V_{D_pm}, respectively. I_{D_pm} and V_{D_pm} are found by computing I_{D_p} and V_{D_p} when $V_{GS} = V_{THN}/2$.

As seen from the figure, there is an overshoot at the early part of the output voltage ($0 \leq t \leq t_1$), that is, the output voltage is greater than the supply voltage. During the overshoot, there is no current flowing from the power supply to the ground. Hence, the short-circuit power consumption is zero during the overshoot. In the figure, t_1 corresponds to the point in time when the output voltage overshoot finishes. t_1 is found by solving the equation $V_{out} = V_{DD}$ using the Taylor series expansion of V_{out} around the point $t = \frac{3}{2}\frac{\tau V_{THN}}{V_{DD}}$.

Referring to Figure 35, in region 2 ($t_n \leq t \leq t_{satp}$), NMOS transistor is saturated and PMOS transistor is in the linear region. t_{satp} represents the point in time when the PMOS transistor enters the saturation region. In this region, PMOS current (I_p) is approximated by a linear function of time as demonstrated in Figure 36 and NMOS current is found using (82). Then, the differential equation of (81) is solved for V_{out}.

At $t = t_{satp}$, PMOS transistor is entering the saturation region. Hence, at time $t = t_{satp}$,the following saturation condition is satisfied

$$V_{out} = V_{DD} - V_{Dp} \tag{84}$$

To find t_{satp}, we use a Taylor series expansion around the point $t = \tau - \frac{\tau}{V_{DD}}(V_{THP} + V_{THN})$

up to the second order coefficient, for both V_{out} and V_{D_p} in (84).

In region 3 ($t_{satp} \leq t \leq t_p$), both transistors are saturated. Hence, the PMOS and NMOS currents are found using (82). We note that t_p corresponds to the point in time when the PMOS transistor enters cutoff region.

From Figure 34, short-circuit current I_{sc} (during a rising input) is expressed as

$$I_{sc} = I_p - I_{C_{gsp}} \tag{85}$$

The current through the capacitance C_{gsp} is given by

$$I_{C_{gsp}} = C_{gsp}\frac{dV_{in}}{dt} = C_{gsp}\frac{V_{DD}}{\tau} \tag{86}$$

5.2.2 Short-Circuit Energy Model

Short-circuit energy dissipation occurs in the interval $[t_2, t_3]$ (as shown in Figure 36) since there is a path from the power supply to the ground in this interval, and is defined as

$$E^r_{sc} = V_{DD}\int_{t_2}^{t_3} I_{sc}dt = V_{DD}\left(\int_{t_2}^{t_{satp}} I_{sc}dt + \int_{t_{satp}}^{t_3} I_{sc}dt\right) \tag{87}$$

As shown in Figure 36, I_{sc} is negative until t_2 (when it becomes zero). In addition, t_3 represents the point in time when I_{sc} becomes zero when the PMOS transistor enters the cutoff region. In computing the first integral of (87), a linear approximation of the PMOS transistor current is used. For the second integral of (87), since the PMOS transistor is in saturation region I_p is found from (82).

The value of t_2 is given by

$$t_2 = t_1 + C_{gsp}\frac{V_{DD}}{S\tau} \tag{88}$$

where S is the slope of the line approximating I_p as function of time ($I_p = S(t - t_1)$). The value of t_3 is found by solving

$$I_{DOP}\left(\frac{\tau}{(V_{DD} - V_{THP})\,V_{DD}}\right)^{alpha_p}\left(t - \frac{V_{THP}\tau}{V_{DD}}\right)^{alpha_p} - C_{gsp}\frac{V_{DD}}{\tau} = 0 \tag{89}$$

We use a Taylor series expansion of the term $(t - V_{THP}\tau/V_{DD})^{alpha^p}$ around the point $t = \frac{t_{satp}}{4} + \frac{3\tau}{4}\left(1 - \frac{V_{THP}}{V_{DD}}\right)$ up to the second coefficient to solve (89) .

Then, the short-circuit energy consumed during a rising edge of input is described by

$$\begin{aligned} E_{sc}^{r} &= \frac{V_{dd}(t_{satp}-t_2)}{2}\left((S\left(t_{satp}+t_2\right)-2St_1)-\frac{2C_{gsp}V_{dd}}{\tau}\right) \\ &-\frac{V_{dd}I_{DOP}}{(V_{dd}-V_{THP})^{\alpha_p}}\left(\left(\frac{V_{dd}t_3}{\tau}-V_{THP}\right)^{\alpha_p+1}-\left(\frac{V_{dd}t_{satp}}{\tau}-V_{THP}\right)^{\alpha_p+1}\right) \\ &-\frac{V_{dd}^2C_{gsp}}{\tau}\left(t_3-t_{satp}\right) \end{aligned} \tag{90}$$

Similarly, the short-circuit energy consumption of a CMOS inverter during the falling edge of the input is calculated by

$$\begin{aligned} E_{sc}^{f} &= \frac{V_{dd}(t_{satn}-t_2)}{2}\left((S\left(t_{satn}+t_2\right)-2St_1)-\frac{2C_{gsn}V_{dd}}{\tau}\right) \\ &-\frac{V_{dd}I_{DON}}{(V_{dd}-V_{THN})^{\alpha_n}}\left(\left(\frac{V_{dd}t_3}{\tau}-V_{THN}\right)^{\alpha_n+1}-\left(\frac{V_{dd}t_{satn}}{\tau}-V_{THN}\right)^{\alpha_n+1}\right) \\ &-\frac{V_{dd}^2C_{gsn}}{\tau}\left(t_3-t_{satn}\right) \end{aligned} \tag{91}$$

where the various symbols (such as t_{satn}, C_{gsn}, and V_{THN}) denote the corresponding parameters for the NMOS transistor.

Then, the average short-circuit current consumed by a CMOS inverter per switching is described by

$$E_{sc} = \frac{E_{sc}^{r}+E_{sc}^{f}}{2} \tag{92}$$

5.3 Model Validation

In this section, we illustrate the validity of the analytical model we derived above by comparing through analytical results to the results of circuit simulations in HSPICE. The comparisons are done for a PCMOS inverter realized in a 0.25 μm CMOS technology.

Two of the primary factors that affect short-circuit energy dissipation are the input transition time τ and the size of the capacitive load C_{out}. In Figure 37, we show a comparison of the analytical and simulation results for the short-circuit energy dissipation of a 0.25 μm inverter as a function of the input transition time for two different values of load capacitance, 200 fF and 50 fF. Here, the supply voltage is 2.5 V and the capacitive load C_{out} is 130 fF for both the analytical model and the simulations. Table 7 shows the MOSFET model parameters used for the analytical model. As seen from the figure, the analytical model results match well with the simulation results. The maximum difference between the analytical and simulation results is 12.2% and the average difference is 7.416%. It

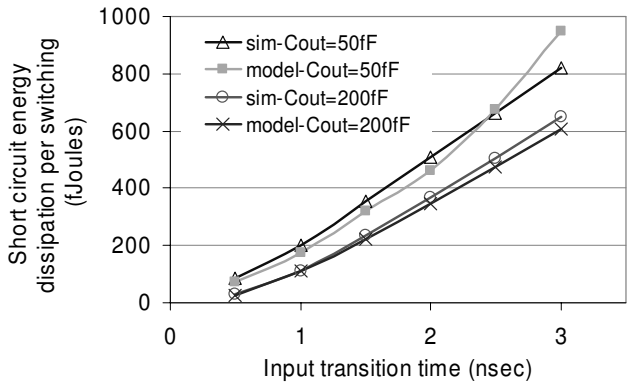


Figure 37: Short-circuit energy dissipation of a 0.25 μm CMOS inverter versus the input voltage transition time.

is also seen from the figure that short-circuit energy dissipation increases as the input transition time increases and C_{out} decreases. For both cases, the difference between the input transition time and output transition (switching) time increases, and hence short-circuit energy dissipation increases.

Table 7: MOSFET model parameters used in the analytical model for the short-circuit energy dissipation of the 0.25 μm CMOS inverter.

	NMOS transistor	PMOS transistor
W (μm)	3.36	6.72
L(μm)	0.25	0.25
α	1.07	1.167
I_{DO} (mA)	1.98	1.87
$\|V_{DO}\|$ (V)	1.17	1.99
$\|V_{TH}\|$ (V)	0.67	0.63
C_{gs} (fF)	6	9
C_{gd} (fF)	7.5	12.5

Figure 38 shows the simulation and analytical results for the 0.25 μm CMOS inverter versus the supply voltage. In addition, the figure illustrates a comparison of our analytical model to the model developed in [20] (which we refer to as the Bisdounis model in the figure). The capacitive load is 150 fF and the input transition time is 2 ns. The average difference of our analytical model results from the simulation results is 5.547% while it is 10.674% for the Bisdounis model. This results from the more accurate MOSFET current model used in our analytical model. This results from the more accurate MOSFET current

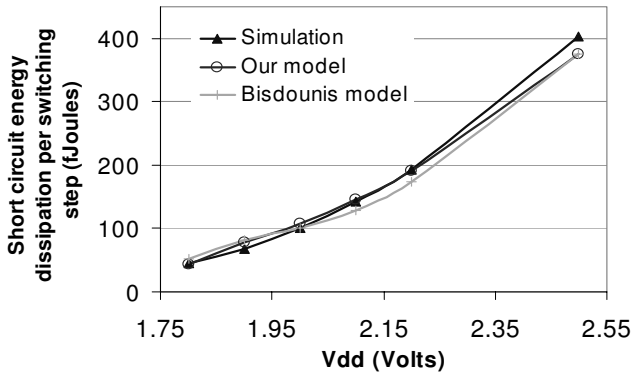


Figure 38: Short-circuit energy dissipation of a 0.25 μm CMOS inverter versus the supply voltage.

model used in our analytical model.

5.4 Energy-Probability Relationship of a PCMOS Switch with the Improved Energy Model

Table 8: Simulation parameters for the fabricated inverter.

Technology		**TSMC 0.25 μm**
Inverter fan-out		7
Load capacitance		130 fF
Nominal Vdd (V)		2.5
Transistor size	$\mathbf{(W/L)}_{pmos}$	6.72 μm/0.24 μm
	$\mathbf{(W/L)}_{nmos}$	3.36 μm/0.24 μm
Vdd (V)		1.5-2.5
σ **(V)**		0.2-0.8
σ_p **(V)**		0.2-0.8
Input rise- and fall-time		2 ns

In Table 8, we show the simulation parameters of an inverter with larger transistors and a larger capacitive load than those considered before in Section 3.3. Further, the rise and fall time of the input signal is higher than the value considered in Section 3.3. In our study, the load capacitance, which is 130 fF (as shown in Table 8) corresponds to the sum of the drain capacitances of the inverter transistors, and the input capacitance of an output driver connected to a chip pin. In Figure 39(a), we show the *E-p* relationship of the inverter (with the parameters shown in Table 8) in the case of input-coupled thermal noise;

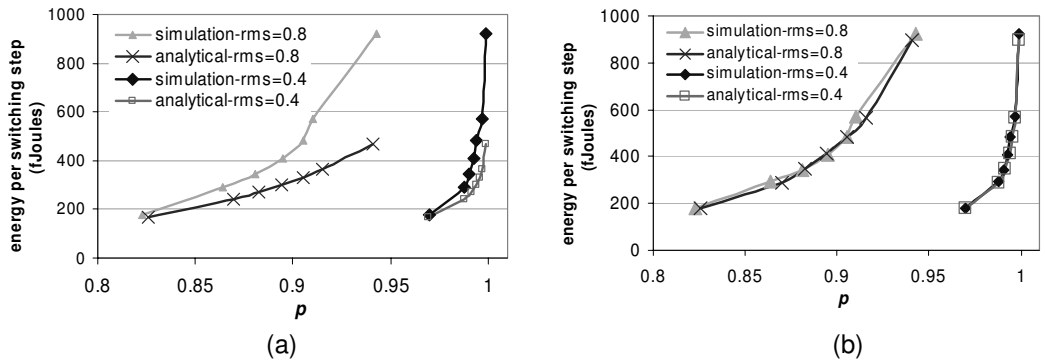


Figure 39: The *E-p* relationship of the CMOS inverter with the parameters shown in Table 8 where the analytical model (a) does not include the short-circuit energy (b) includes the short-circuit energy component.

similar trends and derivations are observed with other noise couplings as well. As seen in Figure 39(a), there is a significant difference between the simulation and analytical results, to be contrasted with the lack of such deviation shown in Figure 8 when the capacitive load C, the transistor sizes, and the rise and fall time of the input pulse are smaller. The significant deviation shown in Figure 39(a) for the 0.25 μm TSMC technology is due to the fact that the analytical model used to produce these results do not account for the short-circuit energy component which is a part of the HSPICE based estimates.

Applying our model to derive short-circuit energy (E_{sc}), the total energy consumed by a CMOS inverter per switching will be

$$E = E_{sc} + E_{sw} \tag{93}$$

where E_{sw} is the switching energy estimated in earlier sections. Using (93) to estimate the energy consumption of the probabilistic inverter (with the parameters shown in Table 8), the resulting *E-p* relationship is shown in Figure 39(b). As seen in the figure, the analytically obtained *E-p* relationship matches extremely well with the *E-p* relationship obtained through simulations. Specifically, the maximum deviation between the analytically modeled and simulated values goes (down) to 2.77% (in Figure 39(b)) when short-circuit energy is accounted from a value of 49.15% when short-circuit energy is not accounted (in Figure 39(a)).

Furthermore, we have also observed in Section 4.3 that when the analytical model including the short-circuit energy is compared against the measurement results, there is a close match between the analytical and the measurement results.

5.5 Conclusions

This chapter has described the short-circuit energy model for the PCMOS switch. The model is based on the alpha-power law MOSFET current model, which is an empirical model describing the current consumption of MOSFET devices. The model was validated against circuit simulations in case when V_{dd} is constant and it was found that the maximum difference between the analytical and simulation results is 12.23 % and the average difference is 7.416 %. When V_{dd} is varied for a 0.25 μm CMOS inverter, the model again follows the simulation results with a difference of 5.547% on average. Furthermore, when this short-circuit energy model is included in the analytical model characterizing the energy-probability relationship of a PCMOS switch, it was seen that the analytical model results for the energy-probability relationship follow the simulation and measurement results closely. It was also seen from the results that short-circuit energy consumption increases as the load capacitance decreases, as the input rise- and fall-times increase, and as V_{dd} increases.

CHAPTER VI

PROBABILISTIC BEHAVIOR OF LARGER PCMOS CIRCUITS

In this chapter, we extend our study of PCMOS circuits beyond the inverter by considering larger logic gates implemented using PCMOS. Unlike the inverter, these gates (e.g., PCMOS XOR) have multiple inputs and multiple nodes where noise can be coupled to. Furthermore, each gate implements a distinct logic function affecting the interaction of the gate with the noise, and hence its probabilistic behavior. As a result, characterizing probabilistic behavior of PCMOS logic gates (when compared to an inverter) involves further considerations. To tackle this problem, we first develop probability models for primitive PCMOS gates. Then, we use these models to derive the probabilities of larger circuits. We use a graph-based model to find the probabilities of larger circuits given the probabilities of their building blocks.

The primary contributions of this work are:

1. A demonstration of a methodology to find the energy-probability behavior of primitive PCMOS logic gates—the primitive gates we have considered are inverter, NAND, NOR and XOR gates.

2. A comparison of the energy-probability relationship of PCMOS logic gates with that of a PCMOS inverter.

3. A methodology to find the energy-probability behavior of PCMOS circuits that are built of PCMOS primitive gates.

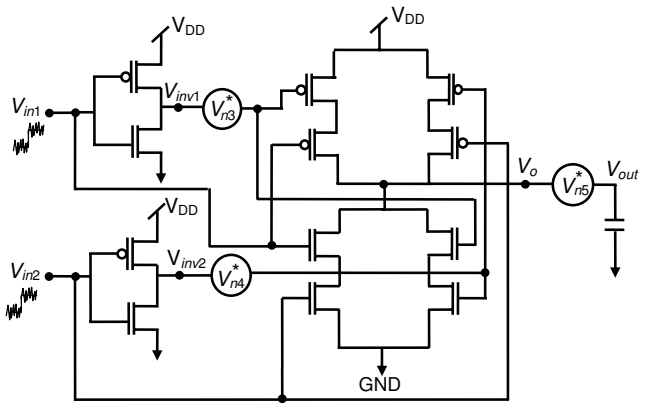


Figure 40: A PCMOS XOR gate coupled with noise at its evaluation nodes.

6.1 Probability and Switching Energy Models of Primitive PCMOS Gates

In modeling the probability parameter p of a PCMOS gate, we consider that noise is coupled to the evaluation nodes [190] of the gate. Evaluation nodes are the nodes that carry the logical information. For example, Figure 40 shows the evaluation nodes of an XOR gate (denoted as V_{in1}, V_{in2}, V_{inv1}, V_{inv2} and V_o). In the figure, noise sources coupled to the nodes V_{inv1}, V_{inv2} and V_o (V_{n3}, V_{n4} and V_{n5}) are also shown. The input signals V_{in1} and V_{in2} are also coupled with noise (not shown in the figure) and the probability of correctness values for these signals are p_1 and p_2, respectively. For simplicity, each noise source is characterized by a Gaussian distribution with mean value of 0. The standard deviation (or rms value) of noise coupled to the inputs V_{in1} and V_{in2} is σ_1 and σ_2. Likewise, the standard deviation of noise coupled to V_{inv1}, V_{inv2} and V_o is σ_3, σ_4 and σ_5, respectively and the probability of correctness values associated these noise sources are p_3, p_4 and p_5. Each probability value p_i ($i \in \{1, 2, 3, 4, 5\}$) is found from (as described in Chapter 3)

$$p_i = 0.5 + 0.5erf\left(\frac{V_{dd}}{2\sqrt{2}\sigma_i}\right) \tag{94}$$

Once we identify the evaluation nodes and the noise sources coupled to them, we construct a probabilistic truth table to find the p of the PCMOS gate. The truth table shows the binary values of the voltages at the input and output nodes of the gate as well as the probabilities associated with these binary values. For example, for the probabilistic XOR gate shown in

Figure 40, we construct the truth table shown in (Table 9). Here, we consider the case when $\sigma_3 = \sigma_4 = \sigma_5$, and hence $p_3 = p_4 = p_5$. This truth table shows the possible combinations of the binary values of the inputs (V_{in1} and V_{in2}) and the output (V_{out}) of the XOR gate. The table also shows the associated probability values of V_{out}, V_{in1} and V_{in2}.

Table 9: A probabilistic truth table for the PCMOS XOR gate.

V_{in1}		V_{in2}		V_{out}	
Val.	Pr.	Val.	Pr.	Val.	Pr.
0	p_1	0	p_2	0	$(-2p_1 - 2p_2 + 6p_1p_2)\,p_3^3 + (3p_1 + 3p_2 - 7p_1p_2)\,p_3^2 + (1 - 3p_1 - 3p_2 + 4p_1p_2)\,p_3 + p_1 + p_2 - p_1p_2$
0	p_1	1	p_2	1	$(2 - 4p_1 - 2p_2 + 6p_1p_2)\,p_3^3 + (-3 + 4p_1 + 3p_2 - 7p_1p_2)\,p_3^2 + (2 - p_1 - 3p_2 + 4p_1p_2)\,p_3 + p_2 - p_1p_2$
1	p_1	0	p_2	1	$(2 - 2p_1 - 4p_2 + 6p_1p_2)\,p_3^3 + (-3 + 3p_1 + 4p_2 - 7p_1p_2)\,p_3^2 + (2 - 3p_1 - p_2 + 4p_1p_2)\,p_3 + p_1 - p_1p_2$
1	p_1	1	p_2	0	$(2 - 4p_1 - 4p_2 + 6p_1p_2)\,p_3^3 + (-1 + 4p_1 + 4p_2 - 7p_1p_2)\,p_3^2 + (-1 - p_1 - p_2 + 4p_1p_2)\,p_3 + 1 - p_1p_2$

Using the probability values associated with V_{out} in Table 9, we find the probability of correctness of a PCMOS XOR gate to be

$$\begin{aligned} p &= \left(\tfrac{3}{2} - 3p_1 - 3p_2 + 6p_1p_2\right)p_3^3 + \left(-\tfrac{7}{4} + \tfrac{7}{2}p_1 + \tfrac{7}{2}p_2 - 7p_1 7p_2\right)p_3^2 \\ &\quad + (1 - 2p_1 - 2p_2 + 4p_1p_2)\,p_3 + \tfrac{1}{4} + \tfrac{p_1}{2} + \tfrac{p_2}{2} \end{aligned} \tag{95}$$

In practical systems, there is always a bandwidth limitation on the noise [216]. For the XOR gate shown in Figure 40, the noise coupled to V_{in1}, V_{in2}, V_{inv1} and V_{inv2} is bandlimited due to the filtering by the gate. In such a bandlimited system, the rms value of the noise is proportional to the bandwidth of the system [134]. Based on this proportionality and using the alpha-power law MOSFET delay model [183], we model the effect of the filtering on the

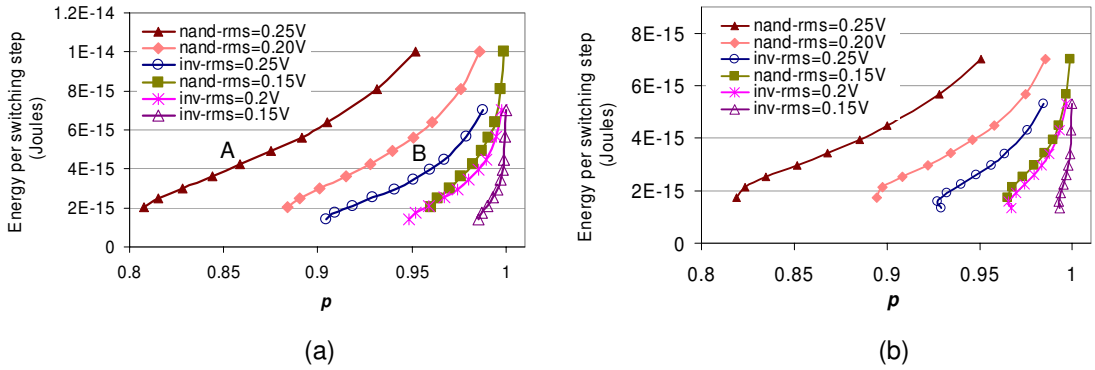


Figure 41: Comparison of energy-probability characteristics of PCMOS NAND and inverter gates in a (a) 90 nm process (b) 65 nm process.

rms value of the noise and computed an equivalent noise rms value observed at the output

$$\sigma_{eq(i)} = \sigma_i \left(\frac{K_{1i} T_{ni}}{\left(K_{2i} T_{ni} + K_{3i} \frac{V_{dd}}{(V_{dd} - V_{TH})^{\alpha}} \right)} \right)^{0.5} \tag{96}$$

where σ_i is the rms value of the noise and T_{ni} is the reciprocal of the maximum frequency component of the noise for $i \in \{1, 2, 3, 4\}$. K_{1i}, K_{2i} and K_{3i} are empirical parameters fitted using HSPICE simulations. V_{TH} and α are the threshold voltage and velocity saturation index parameters of the alpha-power law MOSFET model. Equivalent input probability $p_{eq(i)}$ is found by substituting σ_i in (94) with $\sigma_{eq(i)}$ from (96). Hence, when the bandlimiting effect is considered, the probability of correctness of a PCMOS XOR gate is found by substituting $p(i)$ values in (95) with their corresponding $p_{eq}(i)$ values.

We find the p values for inverter, NAND and NOR gates using the approach outlined for XOR gate above. To model the switching energy consumed by a primitive gate, we use the formula $0.5 C_{eff} V_{dd}^2$, wherein C_{eff} denotes the effective capacitance per switching.

In Figure 41, we show our analytical results showing the energy per switching step (E) versus p of PCMOS NAND and inverter gates realized in 90 nm and 65 nm processes as well as the simulation results validating our analytical models. Different p and E values are found by varying the supply voltage (V_{dd}) of the circuit from 0.15 V to 1 V for both processes. The rms value of the is varied from 0.15 V to 0.25 V. We performed circuit simulations using HSPICE [78].

As seen from Figure 41, given a fixed amount of available noise, energy consumed per

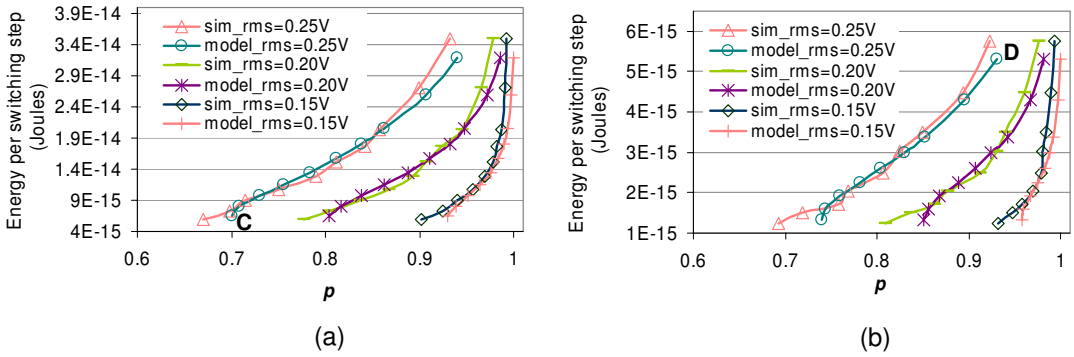


Figure 42: (a) Energy-probability characteristics of a PCMOS XOR gate in a 90 nm process (b) Energy-probability characteristics of a PCMOS XOR gate in a 65 nm process.

switching step of a PCMOS gate increases as its p increases. The rate of increase in E with p grows as p approaches 1. As also seen from the figure, a PCMOS NAND gate displays similar characteristics with a PCMOS inverter in terms of the energy-probability relationship. However, for the same amount of switching energy consumed and given the same rms value of noise, the p of a NAND gate is much smaller than the p of an inverter (for example, points A and B in Figure 41(a)). This results from two reasons: 1) Given a switching energy value E_1, NAND gate consumes E_1 at a supply voltage value of V_{ddn}, while inverter gate consumes E_1 at a supply voltage value of V_{ddi} with $V_{ddi} > V_{ddn}$, because NAND gate has larger capacitances. Since $V_{ddi} > V_{ddn}$, p for the inverter has to be larger than p for the NAND gate. 2) As we move to more complex gates, for the same value of rms and V_{dd}, p is smaller. For example, when rms value of noise is 0.2 V and V_{dd} value is 0.6 V, p is 0.97 for the inverter, while it is 0.92 for the NAND gate. Comparing Figures 41(a) and (b), we observe that both processes display similar characteristics. Below, we compare the processes in detail for an XOR gate.

In Figure 42, we depict our analytical results showing the energy per switching step (E) versus p of a PCMOS XOR gates realized in 90 nm and 65 nm processes as well as the simulation results validating our analytical model. The noise rms value is again varied from 0.15 V to 0.25 V. When we compare Figures 42(a) and 42(b), we see that PCMOS XOR, NAND and inverter gate haves similar energy-probability characteristics. Since, the XOR gate is a more complex gate, for a fixed p value it consumes more energy than NAND and

inverter gates. Furthermore, for the reasons stated above, for the same amount of switching energy consumed and given the same rms value of noise, the p of an XOR gate is smaller than the p of NAND and inverter gates. Comparing Figure 42(a) and (b), we observe that for a fixed value of noise rms and switching energy the XOR gate in 65 nm process has a higher p than the XOR gate in 90 nm process. For example, for the points denoted by C in Figure 42(a), and D in Figure 42(b), E is 5.8 fJ, while p for point C is 0.67 and p for point D is 0.92. This difference between the processes is due to the lower capacitances, hence the lower energy consumption for the 65 nm process at a fixed value of V_{dd}. From here, we deduce that, as the feature sizes decrease, the switching energy investment to obtain a specific p value decreases. However, we note that we do not take the leakage energy and switching activity factor into account. If these are considered, since the static energy consumption increases as the feature sizes decrease, there may not be as much benefit as we observe now for the 65 nm process in terms of its p.

6.2 Algorithm to Find the Probability of PCMOS Circuits

Having described our methodology to model the p for primitive gates, and discussed some of our results, we will now explain how we find the p for larger PCMOS circuits. Given a combinational circuit with k inputs and n outputs, we use the following method to find its probability of correctness. We first map the circuit to its primitive gates—the primitive gates we have considered are inverter, NAND, NOR, and XOR gates. We model the circuit as a directed graph $G = (V, E, m, w)$, where V is the set of gates and E is the set of directed edges connecting these gates. Edge $e_{i,j}$ represents a connection from gate v_i to gate v_j. $m(i)$ is a positive integer used to represent the probability model for the gate v_i—since we have four primitive gates, $1 \leq m(i) \leq 4$, and w_{ij} is the probability of correctness for the signal represented by edge $e_{i,j}$. For a gate v_i, $P(i)$ denotes the set of the probabilities of the inputs of gate v_i. $P_{in} = \{p_{in1}, p_{in2}, \cdots, p_{ink}\}$ denotes the set of probabilities at the primary inputs $\{in_1, in_2, \cdots, in_k\}$ of the circuit.

In Figure 43, we show our probability computation algorithm. In our algorithm, we visit each node in a topological order (line 1) so that the probabilities of gates are propagated

Probability Computation Algorithm
input: graph $G(V, E, m, w)$ and the set P_{in} of input probabilities
output: set of output probabilities
$(P_{out} = \{p_{out}(1), p_{out}(2), \cdots, p_{out}(n)\})$

```
1. T = Topological ordering of gates;
2.  while (T is not empty)
3.      v_i = T.pop;
4.          if (all inputs of v_i are primary inputs) then
5.              p_r(v_i) = compute_pr(m(i), P_in(i));
                (P_in(i) ∈ P_in: set of input probabilities for v_i)
6.          else if (v_i has no primary input) then
7.              p_r(v_i) = compute_pr(m_i, P(e_{j,i}));
                (P(e_{j,i}): set of input probabilities for v_i)
8.          else
9.              p_r(v_i) = compute_pr(m_i, P(e_{j,i}), P_in(i));
                (P(e_{j,i}) ⋃ P_in(i): set of input probabilities for v_i)
10.          for (each edge e(i,k))
11.                 w_ik = pr(v_i);
12.          if v_i has a primary output out_k
13.              p_out(k) = pr(v_i); (1 ≤ k ≤ n)
```

Figure 43: Algorithm to find the p values for the outputs of a combinational CMOS circuit.

in the correct order. For each node in a topological order, we first compute the probability associated with the output of the gate v_i (compute_pr). The function compute_pr first finds the rms values of the noise associated with the input probabilities of the gate from (94). Then, using (96) and (94) compute_pr finds p_{eq} value for each input of the gate ($pr(v_i)$). Using these p_{eq} values, the probability at the output of the gate is computed. If all inputs of a gate v_i are primary inputs (PIs) then compute_pr is a function of $m(i)$ and probabilities of the primary inputs ($P_{in}(i)$) (line 5). If gate v_i has no PIs, then compute_pr is a function of $m(i)$ and the probabilities of the incoming edges of v_i ($P(e_{j,i})$) (line 7). If gate v_i has PIs as well as inputs other than PIs, then compute_pr is a function of $m(i)$ and the probabilities of the incoming edges of m_i, $P(e_{j,i})$ and $P_{in}(i)$ (line 9). The probabilities associated with each outgoing edge of v_i ($e_{i,j}$) are assigned to $pr(v, i)$ (lines 10-11). If v_i is a primary output out_k then $p_{out}(k) = pr(v_i)$ (lines 12-13).

We tested our algorithm for small circuits, such as the four-bit parity and inverter chain circuits shown in Figure 44. In Figure 45, we show the analytical and simulation results for these circuits implemented in a 90 nm process. The average difference between the

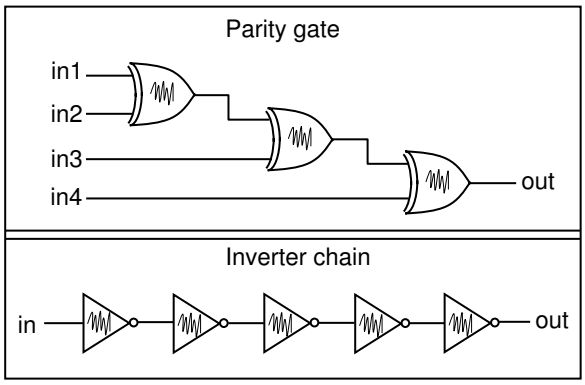


Figure 44: 4-bit parity and inverter chain circuits.

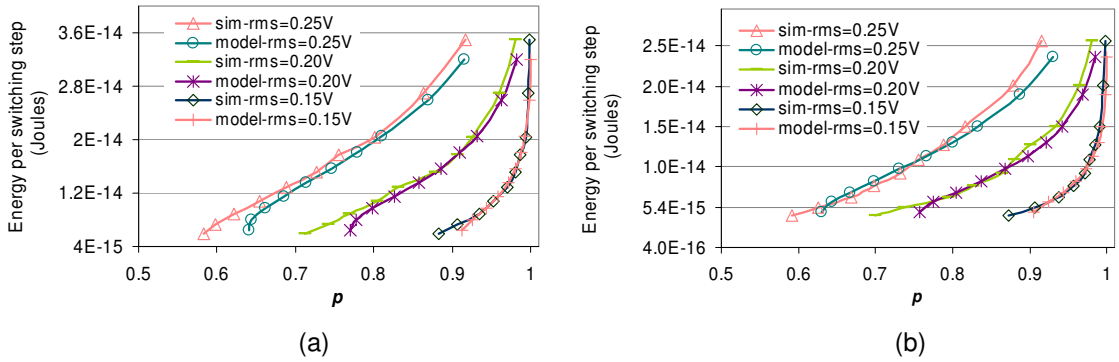


Figure 45: Energy-probability characteristics of (a) 4-bit parity and (b) inverter chain circuits.

analytical and simulated p values is 0.5% for the parity circuit and 0.54% for the inverter chain. The analytical model results deviate the most from the simulation results when V_{dd} is low ($V_{dd} < 0.55$V) because of the underestimation of the circuit delay by our model at low V_{dd} values.

As we move from primitive gates (Figure 42) to larger circuits (Figure 45), the rate of increase in E (with p) is preserved, whereas, for a fixed value of noise rms, values of p in Figure 45 are smaller due to the additive and propagation effect of noise sources in larger circuits. In this context, our analysis is important for design and synthesis of probabilistic circuits to achieve a desired value of p.

6.3 Conclusions

In this work, we have shown a method to derive the energy-probability characteristics of primitive PCMOS gates and presented an algorithm to find the p of PCMOS circuits that are built using these gates. The simulation results for circuits realized in 90 nm and 65 nm processes have demonstrated the validity of our analytical models. Our work provides a basis for analyzing probabilistic behaviors due to noise and other perturbations in future technologies, and can be used in probabilistic design and synthesis methods to improve circuit reliability. For future work, the effect of leakage energy on our analysis can be explored and the library of primitives can be enriched with more complex gates.

CHAPTER VII

ANALYSIS AND OPTIMIZATION OF ENERGY, PERFORMANCE, AND PROBABILITY OF PCMOS CIRCUITS

The downscaling of CMOS device dimensions has enabled an aggressive increase in integration density as described by Moore's Law. However, two other design considerations, noise immunity and power consumption do not scale with Moore's Law. As technology scales down, noise immunity becomes difficult to achieve due to reduced feature sizes and supply voltages, and higher density (see Natori and Sano [149], Shepard [189], van Heijningen *et al.* [217]). The devices are rendered noisy due to effects such as ground bounce and IR drops [128], thermal noise [184], cross-talk noise [47], process variations [24], etc. Therefore, using lower supply voltages to reduce energy consumption reduces noise immunity, and hence the probability of correctness as well. Thus, designing low-energy circuits in the presence of noise is a challenge. To overcome this problem, probabilistic CMOS (PCMOS) based computing has been proposed [155]. In PCMOS based computing, noise is viewed as a resource to achieve low-energy computing rather than an impediment in contrast to the approaches of Kish [100], Hegde and Shanbhag [70], and Solomatnikov *et al.* [195]. The PCMOS approach that was introduced in the former chapters harnesses noise as a resource to implement devices that exhibit probabilistic behavior with a well-characterized probability of correctness (p). For example, a PCMOS inverter is a noisy CMOS inverter. Supply voltage of the circuit, as well as the rms value of the noise determine the p of this inverter.

Lowering the supply voltage of a PCMOS gate decreases its dynamic energy consumption, but also decreases p, which might have an adverse effect on the quality of solution delivered by the application. For example, an error-tolerant application would tolerate an error rate below a certain value [192]. Reducing the supply voltage also decreases the switching

speed of the circuit. Therefore, to meet the performance requirement demanded by the application, threshold voltage may have to be lowered. However, doing so would increase the static energy dissipation due to leakage. In this chapter, we study these trade-offs between the energy, performance, and p of a PCMOS circuits including inverter, NAND, NOR, and XOR gates as well as a one-bit full adder. To do so, we use analytical models for energy, propagation delay, and p. Our analysis considers a PCMOS based application that requires its probability of correctness and performance to be in a predetermined range. By varying the supply voltage (V_{dd}) and the threshold voltage (V_{th}), the analysis is able to optimize the EDP of a PCMOS inverter to meet the application requirements. We also consider the sensitivity of our analysis with respect to variations in temperature, V_{th} and V_{dd}.

We perform circuit simulations using BSIM3 models to verify our analytical models. We study the trade-offs between energy, performance, and p by varying V_{dd} and V_{th}. In particular, given a noise rms value, and various constraints on p, performance, and EDP, we find the values of V_{dd} and V_{th} that optimize EDP or p. We consider the following optimizations: 1) Minimizing the EDP of a PCMOS circuit when its performance is bounded from below and its p is in a range ($p_{min} \leq p \leq p_{max}$). 2) Minimizing the EDP of a PCMOS circuit given its EDP is bounded from above and its p is bounded from below. 3) Maximizing the p of a PCMOS circuit when its performance is bounded from below and EDP is bounded from above.

This chapter is organized as follows. Section 7.1 gives an overview of the prior work on the energy, performance and probability trade-offs. In Section 7.2, we derive analytical models for energy, delay, and p. In Section 7.3, we describe our methods to find the values of V_{dd} and V_{th} that optimize EDP or p of PCMOS circuits under given constraints on p, performance and EDP. In Section 7.4, we show the effect of variations in temperature, V_{th}, and V_{dd} on our analysis. In Section 7.5, we discuss two of the issues we have identified in modeling the energy and p of a PCMOS circuit. Finally, in Section 7.6, we conclude the chapter.

7.1 Background

The trade-offs involving the energy, performance, and p of a circuit were also studied by Hegde and Shanbhag [70]. The primary focus of their work was on deriving information-theoretic lower-bounds on the energy consumption of noisy gates. Their lower-bounds guaranteed reliable computation in the presence of noise. By contrast, in our work, p is an independent design parameter, and its value does not necessarily guarantee reliable computation. Our primary concern is not reliable computing, but being able to compute under the constraints on p imposed by the application.

There has been extensive research on the methods to optimize the energy and delay in strong inversion and sub-threshold regions (see [11, 26, 60, 117]). Various metrics have been proposed to achieve energy and delay optimizations. Gonzalez, Gordon, and Horowitz [60] introduced EDP as a metric to evaluate the energy efficiency of CMOS circuits. Penzes and Martin [163] proposed metrics in the form of $E \cdot D^n$, where energy-delay efficiency index $n \geq 0$ characterizes any feasible trade-off between energy and performance. Hofstee [74] suggested the use of energy-performance ratio, which is the ratio of percent increase in energy per operation to percent increase in performance for various possible design parameters. Markovic *et al.* [117] extended the energy-performance ratio approach to a sensitivity analysis. Their design variables were V_{dd}, V_{th}, and transistor sizes. In our work, we limit ourselves to circuits and architectures that we can optimize at an energy-performance ratio of 1, and we use the EDP metric to show the trade-offs between energy, performance, and p. Our design variables are V_{dd} and V_{th}. For simplicity, we ignore the effect of transistor sizes on these trade-offs.

7.2 Modeling the Energy, Delay, and Probability of Correctness of PCMOS Gates

In this section, we develop simple analytical models that characterize PCMOS gates. Energy, propagation delay, and p of a PCMOS gate are related to its size, supply voltage (V_{dd}), and transistor threshold voltage (V_{th}). For simplicity, we select V_{dd} and V_{th} as our design variables. We refer to a circuit operating under the nominal value of V_{th} as the baseline circuit.

Thus, we have a baseline circuit for each value of V_{dd}. To keep the probability modeling simple, we adjust the transistor sizes such that our baseline circuit is symmetric [215], that is, during a binary transition at the output of the gate, the input and output voltages (V_{in} and V_{out}, respectively) become equal when $V_{in} = V_{out} = \frac{V_{dd}}{2}$. This choice of transistor sizes also ensures that the gate has symmetric rise and fall times. Referring to the voltage at which V_{in} becomes equal to V_{out} as V_m, we observed that V_m differs from $\frac{V_{dd}}{2}$ by only 3% even for the extreme case when V_{th} differs from its nominal value by 47%.

7.2.1 Propagation Delay Model

Based on the delay model from logical effort [204], we model the delay of a gate as a simple product of a process-dependent delay constant τ and a unitless delay element d. τ is the delay of an inverter driving an identical inverter with no parasitics. The unitless delay d includes the delay due to the self-loading of the gate and the delay due to the fanout.

Then, the propagation delay in the subthreshold region is described by

$$t_{g,sub} = \tau d = \frac{K_s V_{dd}}{I_o} e^{\frac{V_{dd}-V_{th}}{n\phi_t}} C_{eff,sub} \tag{97}$$

where n is the body effect coefficient and ϕ_t is the thermal voltage $\frac{kT}{q}$. K_s and I_o are fitted parameters (obtained using circuit simulations performed in HSPICE). $C_{eff,sub}$ is the capacitive load for the gate in the subthreshold region. $C_{eff,sub}$ includes the fanout capacitances as well as the intrinsic capacitances of the gate.

We find the propagation delay of a gate in the strong inversion region using a simple α-power law model [183]

$$t_g = K \frac{V_{dd}}{(V_{dd} - V_{th})^{\alpha}} C_{eff} \tag{98}$$

where K is a parameter fitted using circuit simulations. α is the velocity saturation constant which is also fitted using circuit simulations. C_{eff} is the capacitive load for the gate.

7.2.2 Energy Model

We consider the leakage and switching components of the energy. In modeling the leakage energy, we neglect the gate leakage, pn-junction leakage, and gate-induced drain leakage.

We model only the subthreshold leakage component. Based on the BSIM3 v3.2 [213] equation that models the leakage current, the leakage energy consumed per switching cycle is described by

$$E_L = V_{dd} I_{th} \left(1 - e^{-\frac{V_{DS}}{n\phi_t}}\right) \cdot e^{\frac{V_{GS} - V_{th} - V_{off} + dibl \cdot V_{DS}}{n\phi_t}} \cdot t_{gp} L_{DP} \tag{99}$$

where V_{off} is an empirically determined BSIM3 parameter and *dibl* is the DIBL (Drain Induced Barrier Lowering) factor. The values of V_{off} and *dibl* are derived from curve fitting based on circuit simulations. I_{th} is also an empirical parameter and its value is fitted using circuit simulations. The product $t_{gp} L_{DP}$ denotes the cycle time wherein L_{DP} is the logic depth and t_{gp} is the propagation delay as described by equations (97) and (98). We use a value of 25 for L_{DP}. This value is estimated based on the logic depth of the implementations [29] of the hyper-encryption [45] and probabilistic cellular automata [53] algorithms, as well as the logic depth of a 6-tapped FIR filter [156].

The switching energy is given by

$$E_{SW} = a C_{eff} V_{dd}^2 \tag{100}$$

where a denotes the activity factor. In this work, we assume that a is 40%. This value of a is chosen based on the activity factor of the PCMOS full adders used in the implementation of a 6-tapped FIR filter [156], which are subsequently used in realizing H.264 video decoder [231].

Then, the total energy per switching cycle, and EDP are described by

$$E_T = E_{SW} + E_L \tag{101}$$

$$\text{EDP} = E_T t_{gp} \tag{102}$$

7.2.3 Modeling the Probability of Correctness

We used the methodology we described in Chapter 6 to find the p values of our circuits. As we outlined before, we first found the probabilities for the primitive gates—the primitive gates we have considered are inverter, NAND, NOR, and XOR gates, then we used the algorithm we introduced in Chapter 6 to derive the probabilities for more complex gates that are built using these primitive gates.

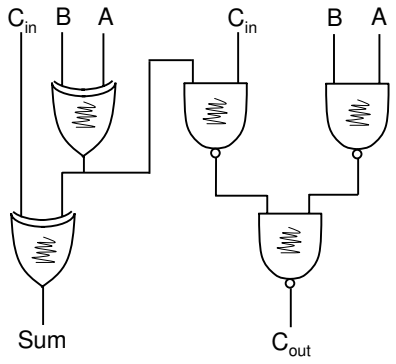


Figure 46: The PCMOS full adder.

7.3 Energy, Performance, and Probability Trade-offs for PCMOS Gates

In this section, we consider the PCMOS full adder shown in Figure 46. We have chosen this implementation for the full adder since we can easily derive its p in terms of the probabilities of primitive PCMOS gates. The XOR and NAND gates used in this full adder are PCMOS gates, each coupled with noise at its evaluation nodes. We consider that this full adder is a primitive element of an architecture realizing a probabilistic application or an error-tolerant application. For example, 240 such full adders are used in realizing an FIR filter [156, 55] that is the building block of the hardware implementing H.264 video decoding [231].

In the following, we employ performance, EDP, and p constraints (imposed by the application) on the PCMOS full adder to derive the optimal V_{th} and V_{dd} operating points that minimize EDP or maximize p of the full adder, for a given rms value of noise. In minimizing EDP, we consider two cases: 1) The full adder is utilized in an architecture block that implements an error-tolerant application. 2) The full adder is used in realizing an architecture block that implements a probabilistic application.

For the architectures implementing error tolerant applications, we minimize EDP under the constraints that EDP needs to be smaller than a maximum value, say, EDP_{max}, and p needs to be greater than p_{min}. The lower-bound on p is due to the quality of the solution delivered by the application. For these applications, we also maximize p to improve the quality of solution under the constraints that EDP needs to be smaller than a maximum value, say, EDP_{max}, and f needs to be greater than f_{min}.

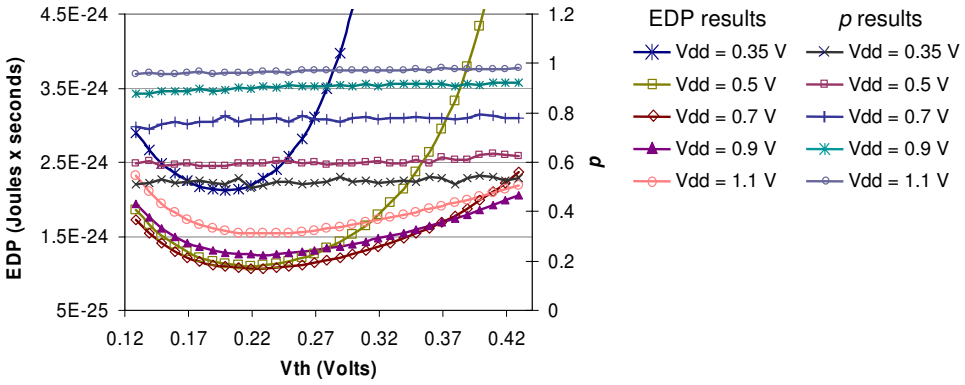


Figure 47: Energy, performance, and probability trade-offs for a PCMOS full adder.

For an architecture block implementing a probabilistic application, we minimize EDP with the following two constraints: (1) The operating frequency of the full adder needs to be greater than a certain value, say f_{min}, and (2) The probability of correctness needs to be in a range, say $p_{min} \leq p \leq p_{max}$. Since p is a design parameter (see [29]) for probabilistic applications, the lower-bound on the probability is deduced from the quality of the solution delivered by the corresponding architecture block, and the upper-bound on the probability is deduced from the quality of the solution as well as the requirement to keep the energy consumption as small as possible.

We note that a study of the trade-offs between the energy, delay and p at the application level is also interesting, however it is beyond the scope of this work. In this section, we primarily use the EDP metric to show the trade-offs between energy, performance, and p of a PCMOS full adder.

In Figure 47, we show the simulation results for EDP and p of a full adder as V_{th} is varied from 0.13 V to 0.43 V at values of V_{dd} varying from 0.35 V to 1.2 V—each curve corresponding to a different value of V_{dd}. The figure shows that at a fixed value of V_{dd}, as V_{th} is decreased, EDP decreases—because of the decrease in delay— until a point is reached where the increase in the leakage energy due to the reduction in V_{th} is dominant and can not be compensated by the decrease in delay. We also observe that, p decreases as V_{dd} is decreased. However the effect of V_{th} on p is negligible. We note that the full range of V_{dd} and V_{th} may not be practical, however, in our analysis we consider that all values are

theoretically possible.

To study the trade-offs evident from Figure 47, we use analytical equations. To compute EDP, we use formulas (97) to (102). To show the trade-offs involving EDP, V_{dd} and V_{th}, we use constant EDP contours in our results below. More specifically, we show contours of normalized EDP (NEDP) (see Figure 48 for example), each denoting the ratio of the minimum EDP (MINEDP) to the EDP corresponding to specific values of V_{dd} and V_{th}.

$$\text{NEDP} = \frac{\text{MINEDP}}{\text{EDP}} \tag{103}$$

As seen from Figure 47, at each value of V_{dd}, EDP as a function of V_{th} bears a minimum value. Thus, to find MINEDP, we find the values of V_{dd} and V_{th} at which EDP is minimized. To this end, we use a two-dimensional search algorithm which iterates over a range of values of V_{dd}: $0.35 \leq V_{dd} \leq 1.2$ V and a range of values of V_{th}: $0.13 \leq V_{th} \leq 0.43$ V. We observed that the result converges when the iteration step for V_{dd} and V_{th} is decreased to 0.0001 V.

The performance of the PCMOS full adder is measured in terms of its maximum switching frequency, denoted as f, and is equal to the reciprocal of its propagation delay t_{gp}. As seen from Figure 46, the full adder produces sum and carry outputs (denoted as "Sum" and "C_{out}," respectively). We consider the propagation delay associated with the carry output as the propagation delay of the full adder, since carry output is propagated (e.g. in a ripple-carry adder), and hence its associated delay determines the delay at the architecture level.

In finding p for the full adder, we combined probabilities of the primitive PCMOS gates (XOR and NAND gates in this case) using the method we described in Chapter 6. We chose the smaller value among the p values for the two outputs of the full adder. Since XOR gate has a smaller p value than a 2-input NAND gate, the probability of correctness associated with output Sum is smaller. Based on these considerations, we computed p of the full adder to be

$$\begin{aligned} p_{adder} &= 6p_{inv}^{11} - 22p_{inv}^{10} + \tfrac{73}{2}p_{inv}^{9} - 33p_{inv}^{8} + \tfrac{109}{8}p_{inv}^{7} + \tfrac{33}{8}p_{inv}^{6} - \tfrac{297}{32}p_{inv}^{5} + \tfrac{51}{8}p_{inv}^{4} \\ &\quad -\tfrac{9}{4}p_{inv}^{3} + \tfrac{1}{4}p_{inv}^{2} + \tfrac{9}{32}p_{inv} + \tfrac{3}{8} \end{aligned} \tag{104}$$

where p_{inv} is the probability of correctness for an inverter gate as given by (39). As seen from this equation and Figure 47, p is primarily determined by V_{dd}. We note that, we have also considered the case that the p associated with the carry bit may be propagated (eg. in a ripple-carry adder), and because of this p of the full adder may be equal to the p of the carry bit. However, our results showed that p for the sum output is significantly worse than the p associated with the carry output. Hence, we decided on using p of the sum output as the p of the full adder.

Below, we formulate the optimization problems we have considered, and describe the algorithms we have developed to solve the problems.

7.3.1 Minimizing EDP under Probability and EDP Constraints

In this section, we consider the problem of minimizing EDP given an upper-bound on EDP and a lower bound on p. Such an optimization would be useful to find optimal circuits for error-tolerant applications.

Figure 48 shows the constant NEDP, performance and p contours for a PCMOS full adder coupled with noise at its evaluation nodes. The noise is characterized by a Gaussian distribution with rms value of 0.1 V. In the figure, contours of constant NEDP are represented by the round curves, contours of constant p are represented by the horizontal lines, and contours of constant frequency (or performance) are represented by the sloped lines. As seen from the figure, NEDP is high at lower values of V_{dd} and V_{th}. However, for very low values of V_{th}, NEDP becomes smaller due to the increased leakage energy. Similarly, for very low values of V_{dd}, NEDP becomes smaller due to the increased delay. We note that, the NEDP curves have small kinks near the border of subthreshold region, which are due to the discontinuity of the analytical delay model (equations (97) and (98)) at the boundary of subthreshold region. Figure 48 also shows that p increases as V_{dd} increases. It is also seen from the figure that the higher the value of V_{dd} with respect to V_{th}, the higher the performance.

Given these trade-offs, our objective is to find the optimal V_{dd} and V_{th} operating points that minimize the EDP within the given constraints. The problem is stated as follows.

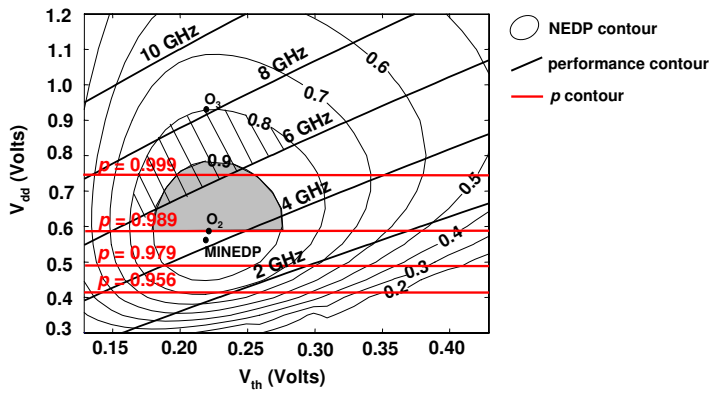


Figure 48: Analytically derived constant NEDP, performance, and p contours for a PCMOS full adder coupled with noise having an rms value of 0.1 V.

Minimize:

$$\begin{aligned} \text{EDP} = \quad & V_{dd} I_{th} \left(1 - e^{-\frac{V_{dd}}{n\phi_t}}\right) \cdot e^{\frac{-V_{th} - V_{off} + dibl \cdot V_{dd}}{n\phi_t}} t_{gp}^2 L_{DP} \\ & + a C_L V_{dd}^2 t_{gp} \end{aligned} \tag{105}$$

subject to:

$$\text{EDP} \leq \text{EDP}_{max} \tag{106}$$

$$p \geq p_{min} \tag{107}$$

To solve this problem, we use the two-dimensional search algorithm described in Figure 49. Our search is confined in the NEDP contour corresponding to EDP_{max} and the area above the probability contour corresponding to p_{min}. For example, if the upper-bound on EDP is $\frac{\text{MINEDP}}{0.9}$, and the lower-bound on p is 0.989, then the feasible region for the search is the gray shaded area shown in Figure 48. The search across the V_{dd} values starts at the value of V_{dd} ($V_{dd\text{MINEDP}}$) at which MINEDP—the global minimum for EDP—is observed. At every iteration over V_{dd}, the value of V_{dd} is increased. The search across the V_{th} values starts at the value of V_{th} ($V_{th\text{MINEDP}}$) at which MINEDP is observed, and the search is performed in two different directions until the EDP contour corresponding to EDP_{max} is reached: with positive steps of V_{th} (moving towards right in Figure 48), and with negative steps of V_{th} (moving towards left in Figure 48). This particular choice of the initial values of V_{dd} and V_{th} and searching across increasing values of V_{dd} are due to following reasons: (1) From the definition of NEDP as shown by equation (103), on the V_{dd}-V_{th} plane, the point with EDP value of MINEDP (also shown in Figure 48) is inside every possible NEDP contour. (2) If the

point corresponding to MINEDP is already in the feasible area for the search, that is, in the area confined by the NEDP contour for EDP_{max} and the probability contour for p_{min} then EDP is minimized at this point, and there is no need to search in the region under $V_{ddstart}$. (3) If $V_{ddstart}$ is smaller than the lower limit on V_{dd} (V_{ddmin}) that is determined by p_{min}, then V_{dd} value is increased until V_{ddmin} is reached (lines 8 and 24 in Figure 49). With this as background and as seen from Figure 49, for the given values of p_{min}, EDP_{max}, and noise rms, the algorithm can be described as follows:

1. We initialize the values of V_{dd} and V_{th} to $V_{ddstart}$ and $V_{thstart}$, respectively.
2. We assign a sufficiently large value to EDP_{min}.
3. We compute V_{ddmin} using the probability constraints as defined by (107), (104), and (39).
4. We increase V_{dd} from $V_{ddstart}$ in sufficiently small steps until we reach a V_{dd} value at which (106) is not satisfied. For each V_{dd} that is greater than V_{ddmin}:
   (a) We assign a sufficiently large value to the minimum value of the EDP of this step ($\text{EDP}_{min}(i)$).
   (b) We initialize V_{thL} and V_{thR} to $V_{thstart}$, where V_{thL} denotes the V_{th} iterator towards left and V_{thR} denotes the V_{th} iterator towards right.
   (c) We increase V_{thL} and V_{thR} in sufficiently small steps until (106) is not satisfied. For each value of V_{thL} and V_{thR}:
      i. Using the current values of V_{dd} ($V_{dd}(i)$) and V_{thR} ($V_{thR}(j)$), and equation (105), we compute EDP_{R}. We compute EDP_{L} similarly.
      ii. If EDP_{R} is smaller than $\text{EDP}_{min}(i)$, then we update the values of $\text{EDP}_{min}(i)$, the optimum value of the V_{th} at this step ($V_{thopt}(i)$), and the optimum value of V_{dd} at this step ($V_{ddopt}(i)$) as shown in the lines 12-14 of the pseudocode. We repeat this step for EDP_{L} and V_{thL} (lines 15-17).
   (d) If $\text{EDP}_{min}(i)$ is smaller than EDP_{min}, then we update the values of EDP_{min}, V_{ddopt}, and V_{thopt} as shown in lines 21-23 of the pseudocode.

```
1.  Start: Input: p_min, EDP_max, noise RMS and S >> 1
2.         V_thstart = V_thMINEDP; V_thstep = V_thstart / S;
3.         V_ddstart = V_ddMINEDP; V_ddstep = V_ddstart / S;
4.         i = 0; j = 0; EDP_min = 1; V_dd(0) = V_ddstart;
5.         compute V_ddmin using p_min, (12) and (9);
6.  V_dd Loop: repeat
7.             EDP_min(i) = 1; V_thL(0) = V_thR(0) = V_thstart;
8.             if V_dd(i) > V_ddmin
9.  V_th Loop:     repeat
10.                compute EDP_R using V_dd(i), V_thR(j), and (13);
11.                compute EDP_L using V_dd(i), V_thL(j), and (13);
12.                    if EDP_R < EDP_min(i)
13.                        EDP_min(i) = EDP_R;
14.                        V_thopt(i) = V_thR(j); V_ddopt(i) = V_dd(i);
15.                    if EDP_L < EDP_min(i)
16.                        EDP_min(i) = EDP_L;
17.                        V_thopt(i) = V_thL(j); V_ddopt(i) = V_dd(i);
18.                    j = j+1; V_thL(j) = V_thL(j-1)- V_thstep;
19.                    V_thR(j) = V_thR(j-1)+ V_thstep;
20.                until EDP > EDP_max;
21.                if EDP_min(i) < EDP_min
22.                    EDP_min = EDP_min(i);
23.                    V_ddopt = V_ddopt(i); V_thopt = V_thopt(i);
24.            i = i+1; V_dd(i) = V_dd(i-1)+ V_ddstep;
25.        until EDP > EDP_max;
26.        report EDP_min, V_ddopt, and V_thopt;
```

Figure 49: The pseudocode for the algorithm to find the optimal V_{dd} and V_{th} values that minimize EDP under probability and EDP constraints.

Our results have shown that if V_{ddmin} is greater than $V_{ddstart}$, then the optimal V_{dd} value is very close to V_{ddmin}. Referring to Figure 48, for example, if the probability constraint is such that $p \geq 0.989$ corresponding to a V_{ddmin} of 0.59 V, and the EDP constraint is $\text{EDP} \leq \frac{\text{MINEDP}}{0.9}$, then the algorithm finds the optimal values of V_{dd} and V_{th} as 0.59 V, and 0.22 V, respectively. The point corresponding to these values of V_{dd} and V_{th} is denoted as O_2 in the figure. We note that the point corresponding to MINEDP has a V_{dd} value of 0.57 V which is smaller than V_{ddmin}. On the other hand, if V_{ddmin} is smaller than $V_{ddstart}$, then the optimal values of V_{dd} and V_{th} are the same as $V_{dd\text{MINEDP}}$ and $V_{th\text{MINEDP}}$, respectively.

7.3.2 Maximizing p under EDP and Performance Constraints

In this section, we study the problem of maximizing the p given an upper-bound on EDP and a lower-bound on performance. Error-tolerant applications for which there is a maximum limit on the error would benefit an optimization solving this problem. The problem is formulated as follows:

```
1.  Start:  Input: EDP_max, f_min, noise RMS, and S >> 1
2.          V_thstart = V_thMINEDP; V_thstep = V_thstart / S;
3.          compute V_ddstart using V_thstart, (17), (3), and (4);
4.          V_ddstep = V_ddstart / S;
5.          i = 0; j = 0; p_maxR = 0; p_maxL = 0; V_thR(0) = V_thL(0) = V_thstart;
6.  V_th Loop:  repeat
7.                  compute V_ddR(0) using V_thR(i), (17), (3), and (4);
8.                  p_maxR (i) = 0;
9.  V_dd Loop:      repeat
10.                     compute p_R using V_ddR(j), (12), and (9);
11.                     if p_R > p_maxR(i)
12.                         p_maxR(i) = p_R; V_thopt(i) = V_thR(i); V_ddopt(i) = V_ddR(j);
13.                 j = j+1; V_ddR(j) = V_ddR(j-1)+ V_ddstep;
14.                 compute EDP_R using V_ddR(i), V_thR(j), and (13);
15.             until EDP_R > EDP_max;
16.             if p_maxR (i) > p_maxR
17.                 p_maxR = p_maxR(i); V_ddoptR = V_ddopt(i); V_thoptR = V_thopt(i);
18.             i = i+1; V_thR(i) = V_thR(i-1)+ V_thstep;
19.         until EDP_R > EDP_max;
20.         repeat steps 6 to 19 with decreasing steps of V_th;
21.         report p_maxR, V_ddoptR, V_thoptR, p_maxL, V_ddoptL, V_thoptL;
22.     if p_maxL > p_maxR
23.         p_max = p_maxL; V_ddopt = V_ddoptL, V_thopt = V_ddoptL;
24.     else
25.         p_max = p_maxR; V_ddopt = V_ddoptR, V_thopt = V_ddoptR;
26.     report p_max, V_ddopt and V_thopt;
```

Figure 50: The pseudocode for the algorithm to find the optimal V_{dd} and V_{th} values that maximize p under performance and EDP constraints.

Maximize:

$$p$$

subject to:

$$\text{EDP} \leq \text{EDP}_{max} \tag{108}$$

$$f \geq f_{min} \tag{109}$$

To find the values of V_{dd} and V_{th} that maximize p under the given constraints, we use the search algorithm shown in Figure 50. Using this algorithm, the search is performed in a region to the left of the performance contour corresponding to the performance limit f_{min} and enclosed by the NEDP contour corresponding to the EDP limit EDP_{max}. For example, if f_{min} is 6 GHz and EDP_{max} is $\frac{\text{MINEDP}}{0.8}$, then the feasible area for the search will be the hatched area shown in Figure 48. The iterations over V_{th} are performed in two directions, that is, the iterations start at fixed value of V_{th} and proceed with increasing, as well as decreasing values of V_{th}. In Figure 50, lines 6-19 describe the algorithm steps when the search is performed with positive steps of V_{th}, and line 20 states that the steps 6-19 should be

repeated with negative steps of V_{th}. Another approach to do the search in the V_{th} dimension is to first compute the V_{th} values at the two points where the NEDP contour corresponding to EDP_{max} intersects with the performance contour corresponding to f_{min}, and iterate across V_{th} in the region between the V_{th} values of the intersection points. We adopt the bidirectional approach since the bidirectional approach delivers the same complexity with the approach in which one computes the intersection points. In this way, we can avoid the iterative computations of the intersection points. Due to this bidirectional search over V_{th}, we find two different maximum values of p (denoted as p_{maxL} and p_{maxR}) and compare them finally to find the maximum value of p (p_{max}). The algorithm (also shown in Figure 50) is as follows:

1. We initialize V_{th} to $V_{th\text{MINEDP}}$. We denote the initial value of V_{th} by $V_{thstart}$. Using $V_{thstart}$, and the performance constraint, we compute the initial value of V_{dd}. This initialization guarantees that the search starts at a point in the feasible search region.

2. We assign sufficiently small values to p_{maxL} and p_{maxR}.

3. We initialize V_{thR} (V_{thR} denotes the right-hand side iterator for the search over V_{th}) to $V_{thstart}$.

4. We increase V_{thR} in sufficiently small steps until we reach a V_{dd} value at which (108) is not satisfied. For each value of V_{thR}:

   (a) We assign a sufficiently small value to the value of p_{maxR} at this iteration step ($p_{maxR}(i)$).

   (b) We compute the initial value of V_{ddR} ($V_{ddR}(0)$) for the current value of V_{thR} using (97) and (98), and given (109).

   (c) We increase V_{ddR} from $V_{ddR}(0)$ in sufficiently small steps until we reach a V_{ddR} value at which (108) is not satisfied. For each value of V_{ddR}:

      i. Using the current value of V_{ddR} ($V_{ddR}(j)$), and equations (104) and (39), we compute the probability value (p_R) at this step.

ii. If p_R is greater than $p_{maxR}(i)$, then we update the values of p_{maxR}, $V_{thopt}(i)$, and $V_{ddopt}(i)$ as shown in line 12 of the pseudocode.

(d) If $p_{maxR}(i)$ is larger than p_{maxR}, then we update the values of $p_{maxR}(i)$, V_{ddoptR}, and V_{thoptR} as shown in line 17 of the pseudocode.

5. We repeat the steps described above for the left-hand side iterations over V_{th}, and find the values of p_{maxL}, V_{ddoptL}, and V_{thoptL}.

6. Comparing p_{maxL} and p_{maxR}, we find p_{max} and the corresponding optimal values V_{ddopt} and V_{thopt}.

Our results have shown that in case when we maximize p under the EDP constraint $\text{EDP} \leq \text{EDP}_{max}$, and the performance constraint $f \geq f_{min}$, the optimal V_{dd} and V_{th} values are found to be on the NEDP contour corresponding to EDP_{max}. Furthermore, denoting the points on the contour by the tuples (V_{th}, V_{dd}), the optimal values of V_{th} and V_{dd} are observed at the point where the V_{dd} component of the tuple (V_{th}, V_{dd}) on the $\text{NEDP} = \text{NEDP}_{max}$ contour reaches its maximum value. For example, if the performance constraint is such that $f \geq 6$ GHz, and the EDP constraint is $\text{EDP} \leq \frac{\text{MINEDP}}{0.8}$, then the algorithm finds the optimal values of V_{dd} and V_{th} as 0.93 V, and 0.22 V, respectively. As shown in Figure 48, this point, denoted as O_3 is on the $\text{NEDP} = 0.8$ contour, and the V_{dd} component of this point is the largest among the other points on this NEDP contour.

7.3.3 Minimizing EDP under Performance and Probability Constraints

In this section, we consider the problem of minimizing EDP given a lower-bound on the performance and an interval of values for p. The problem is stated as follows.

Minimize:

$$\text{EDP}$$

subject to:

$$f \geq f_{min}$$

$$p_{min} \leq p \leq p_{max}$$

```
1.  Start:  Input: p_min, p_max, f_min, noise RMS, and S >> 1
2.          V_thmin = 0.12; V_thstep = V_thmin / S; p_step = p_min / S;
3.          I = 0; j = 0; EDP_min = 1; p(0) = p_min;
4.  p Loop:     repeat
5.                  compute V_dd(i) using p(i), (12), and (9);
6.                  compute V_thmax using f_min , (3), and (4);
7.                  EDP_min(i) = 1; V_th(0) = V_thmin;
8.  V_th Loop:      repeat
9.                      compute EDP using (13);
10.                     if EDP < EDP_min(i)
11.                         EDP_min(i) = EDP;
12.                         V_thopt(i) = V_th(j);  V_ddopt(i) = V_dd(i);
13.                          j = j+1; V_th(j) = V_th(j-1)+ V_thstep;
14.                 until V_th(j) > V_thmax;
15.                 if EDP_min(i) < EDP_min
16.                     EDP_min = EDP_min(i);
17.                     V_ddopt = V_ddopt(i); V_thopt = V_thopt(i);
18.                 i = i+1; p(i) = p(i-1) + p_step;
19.             until p(i) > p_max;
20.             report EDP_min, V_ddopt and V_thopt;
```

Figure 51: The pseudocode for the algorithm to find the optimal V_{dd} and V_{th} values that minimize EDP under performance and probability constraints.

The pseudo-code of the two-dimensional search algorithm is described in Figure 51. As seen from the figure, the algorithm is very similar to the previous two algorithms. For brevity, we will skip describing the details of this algorithm.

In Figure 52, we show the NEDP, performance, and probability contours for a PCMOS full adder coupled with noise having rms value of 0.2 V. In this figure, for example, if the performance constraint is set at 6 GHz, the search algorithm searches for the optimal V_{dd} and V_{th} operating points in the region to the left of the 6 GHz line. Furthermore, if p_{min} and p_{max} are 0.8 and 0.9, respectively, then the search is performed within the shaded area shown in the figure. The algorithm finds that for the optimal EDP point, the values of V_{dd} and V_{th} are 0.72 V and 0.22 V, respectively, as shown by the point O_1 in Figure 52. As seen from Figure 52, the supply and threshold voltages for optimal EDP are closer to the probability contour $p = p_{min}$, that is, operating the PCMOS full adder at lower supply voltages is more preferable in terms of the EDP. However, the supply voltages can not be reduced further beyond the point where the probability constraint is not satisfied anymore.

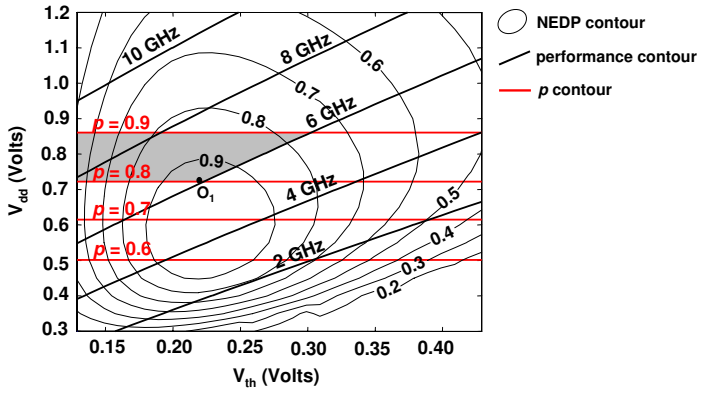


Figure 52: Analytically derived constant NEDP, performance, and p contours for a PCMOS full adder coupled with noise having an rms value of 0.2 V.

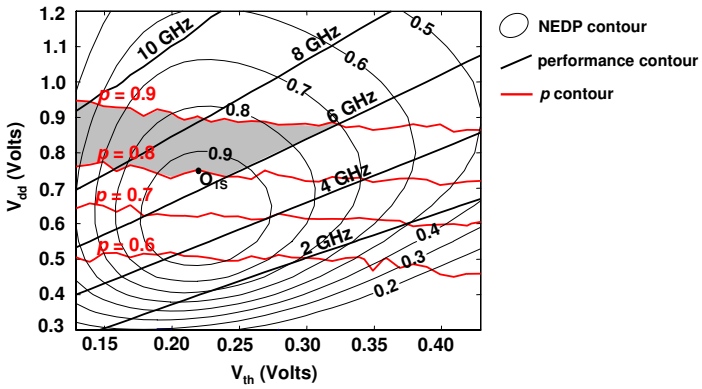


Figure 53: Constant NEDP, performance, and p contours from circuit simulations for a PCMOS full adder coupled with noise having an rms value of 0.2 V.

7.3.4 Simulation Results

In this section, we validate our analytical models through comparing our analytical results with the simulation results for EDP, performance, and p.

We performed circuit simulations in HSPICE using BSIM3 models for a CMOS inverter in a 0.13 μm process to measure the PCMOS full adder's static and dynamic energy consumption, propagation delay, and p. In measuring the propagation delay of the full adder, we considered the worst case delay among the delays associated with every possible binary transition that might occur at the inputs of the full adder. We measured the static energy and the switching energy separately. In measuring the switching energy, we used random strings of binary values at the inputs of the full adder.

In modeling the thermal noise that is coupled to the evaluation nodes of the full adder,

the noise source is assumed to be a random process characterized by a Gaussian distribution. The details of modeling the noise, the coupling of the noise and calculation of p in the circuit simulations for a PCMOS inverter were described in Chapter 3.

The results of the simulations for a PCMOS full adder coupled with noise having an rms value of 0.2 V are shown in Figure 53. When compared to the analytical results shown in Figure 52, we observe that shapes and locations of the NEDP and performance contours are quite similar. For small values of EDP, the NEDP contours shown in Figure 53 are slightly narrower across V_{dd} values compared to the NEDP contours in Figure 52. For example, the contours corresponding to NEDP = 0.6 are narrower in Figure 53 than they are in Figure 52. This difference between the analytical and simulated NEDP contours is due to the inaccuracy of the analytical model in estimating the propagation delay and the leakage power. Referring to the ratio of the absolute difference between the simulation result and the analytical result to the simulation result as the error, we find that the average error for the analytical delay model is 3.74%, and for the analytical energy model it is 3.32%. These errors lead to an error of 4.44% on the average for the EDP. The standard deviation of the error for the analytical EDP results is only 4.21%.

Since we used a linear delay model based on the logical effort, our model does not fully consider the effects of the topology of a gate or the inputs of a gate on the delay. Hence, the analytical delay results differ from the simulation results as seen from comparing the performance contours in Figure 52 and Figure 53. We observe that the analytical model slightly underestimates the propagation delay for low values of V_{th}, and slightly overestimates it for high values of V_{th}.

Comparing the p contours of Figure 52 and Figure 53, we observe that the p contours found using simulations are traversing higher values of V_{dd}. This is caused by the fact that the transistors of the XOR gates used in simulations are not symmetrical, whereas the analytical model considers the case when the transistors are symmetrical. As we explained in Chapter 3 we have a more accurate probability model for the case when the transistors are not symmetrical. However, the more accurate model requires the midpoint voltage of the CMOS inverter, which we have not derived in the subthreshold region. Thus, we have

chosen to use the probability model in (39) for simplicity in this work. Furthermore, in Figure 53, the p contours are not exactly horizontal, but have a negative slope (which is very small in magnitude). This weak dependency of p on V_{th} is due to the dependency of p on the midpoint voltage of the inverter.

Comparing Figures 52 and 53, we see that the feasible search regions (the areas shaded with gray) for the optimization example provided at the end of Section 7.3.3 slightly differ. Furthermore, optimal values of V_{dd} and V_{th} found from simulations are 0.75 V and 0.22 V (denoted as O_{1s} in Figure 53) as opposed to 0.72 V and 0.22 V found using the analytical models (denoted as O_1 in Figure 52). We note that to find the optimal operating points in case of simulations, we use a variant of the algorithm described in Figure 51. We replace the steps for calculations of $V_{dd}(i)$, V_{thmax}, and EDP by search steps. The search step traverses the simulation results, and finds the closest values for $V_{dd}(i)$, V_{thmax}, and EDP.

7.4 Process and Operating Point Variations

So far, in our analysis we have not considered any variations in threshold voltage, supply voltage, and the operating temperature. However in reality, threshold voltage might change due to process variations and changes in the operating temperature. In addition, chip temperature changes due to heat dissipation. Furthermore, supply voltage might change due to IR voltage drop, or simultaneous switching noise (SSN) [128]. Neglecting the coupling between the chip temperature and power dissipation [11], in this section, we demonstrate the impact of the variations in temperature, V_{th}, and V_{dd} on the energy, performance, and p of PCMOS circuits.

7.4.1 Variations in Temperature

With increasing circuit density, die area, and on-chip power dissipation, the variation of temperature across an integrated circuit becomes significant. The operating temperature of a circuit can be anywhere between 25°C and 125°C. The threshold voltage at temperature T can be calculated [11] using

$$V_{thT} = V_{th} - k\,(T - T_{amb}) \tag{110}$$

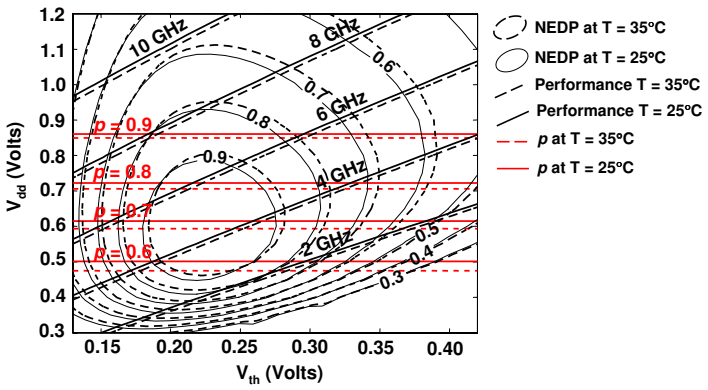


Figure 54: Analytically derived constant NEDP, performance, and p contours for a PCMOS full adder at temperatures $T = 25°\text{C}$ and $T = 35°\text{C}$.

where V_{thT} is the threshold voltage at temperature T, T_{amb} is the ambient temperature (25°C), and k is the threshold voltage temperature coefficient whose typical value for a 0.13 μm process is 0.7 mV/K [207].

Temperature variation also affects the rms value of the noise, since we consider a thermal noise source. For simplicity, we assume that the noise source is a resistive noise source and therefore, we calculate the rms value of noise at temperature T using

$$\text{rms}_T = \text{rms} \cdot \sqrt{\frac{T}{T_{amb}}} \tag{111}$$

Figure 54 shows the effect of increasing temperature from 25°C to 35°C on a PCMOS full adder. At the ambient temperature, the rms value of the noise coupled to the full adder is 0.2 V. In the figure, the dashed contours correspond to the results when T is 35°C and the solid contours correspond to the results when T is 25°C. As seen from the figure, NEDP and performance contours are shifted to right when temperature is increased. This results from the decrease in V_{th} due to the increase in temperature. Furthermore, p contours are shifted higher in the V_{dd} domain, that is, to obtain the same value of p, a higher V_{dd} value is required at a higher temperature. Hence, at a fixed value of V_{dd}, p decreases as T increases. This decrease in p is due to the increased rms value of noise because of the increased temperature. Resulting from these variations in NEDP, performance, and p, optimal values of V_{dd} and V_{th} also change. For example, when we consider minimizing the EDP of a full adder with probability constraint $0.80 \leq p \leq 0.90$, and performance constraint

$f \geq 6$ GHz, optimal values of V_{dd} and V_{th} at $T = 35°$C are 0.73 V and 0.22 V as opposed to the values of 0.72 V and 0.22 V at $T = 25°$C found previously in Section 7.3.3. In this example the change in optimal values of V_{dd} and V_{th} is negligible, however, if T is increased even further, we observe significant changes in the optimal values. For example, when the same problem is considered at $T = 85°$C, optimal V_{dd} and V_{th} values become 0.86 V and 0.24 V, corresponding to 19.4% and 9.1% difference when compared to the original values of 0.72 V and 0.22 V for $T = 25°$C.

7.4.2 Variations in Threshold Voltage

Threshold voltage variations, caused by the underlying process variations might have a significant effect on the circuit performance and energy consumption [24]. Empirical evidence suggests that variations in V_{th} can be modeled by a Gaussian distribution [145]. So, we model the threshold voltage variations using a Gaussian distribution. The mean value of the distribution is 0 and its standard deviation σ_{Vth} is equal to 10% of the threshold voltage [27].

We note that threshold voltage variations result from process variations such as the variations in the channel length, fluctuations in the channel doping, and variations in the oxide thickness. Variations in these process parameters affect the performance and energy not only through the threshold voltage but also through other factors. For example, the "ON" current is inversely related to the oxide thickness and the channel length, and the "ON" current is one of the factors that determine the performance and switching energy. For simplicity we ignore the impact of these individual variations on performance and energy. Further, we ignore the correlations between these variations for individual transistors.

Figure 55 depicts the effect of the variations in V_{th}. In the figure, the dashed contours correspond to the results when there is variation in V_{th}, and the solid contours correspond to the results when there is no variation in V_{th}. We only show the NEDP and performance contours and ignore the effect of V_{th} on p. As seen from the figure, the NEDP contours are shifted upwards, that is, for a fixed value of V_{th}, to obtain the same value of NEDP, a higher V_{dd} value is required. This results from the fact that the change in EDP is larger when there

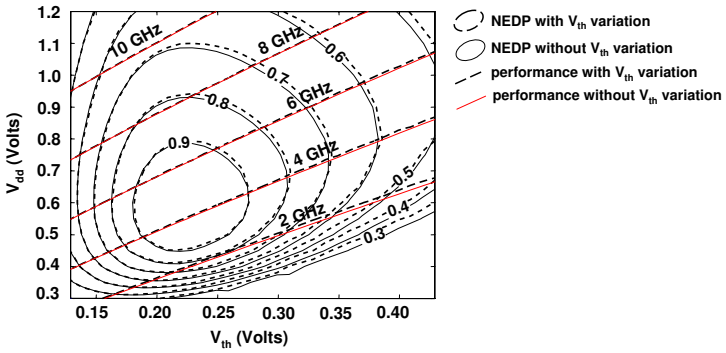


Figure 55: Analytically derived constant NEDP and performance contours for a PCMOS full adder with V_{th} variations when mean values of EDP and performance are considered.

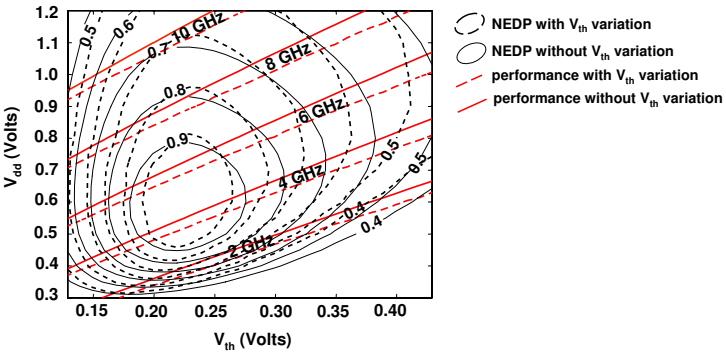


Figure 56: Analytically derived constant NEDP and performance contours for a PCMOS full adder with V_{th} variations when mean value plus one standard deviation is considered for EDP and performance.

is a positive change in V_{th} (e.g. $V_{th} + \sigma$) compared to a negative change in V_{th} of the same magnitude (e.g. $V_{th} - \sigma$). We note that, in Figure 55, we only show the mean values of NEDP and performance. As a result, the differences in NEDP and performance seem to be very small. Due to averaging, the kinks in the NEDP contours have also disappeared.

If we consider a worst case scenario such as the mean plus one standard deviation, that is, we find the mean plus one standard deviation of EDP and performance, then the effect of V_{th} variations on EDP and performance is more significant as shown in Figure 56. Furthermore, the optimal values of V_{dd} and V_{th} differ from the optimal values found when there is no V_{th} variation. For example, when we consider minimizing the EDP of a full adder with the probability constraint $0.80 \leq p \leq 0.90$, and performance constraint $f \geq 6$ GHz, the optimal values of V_{dd} and V_{th} are now 0.73 V and 0.23 V corresponding to 1.4% and 4.5% difference compared to the optimal values of 0.72 V and 0.22 V obtained when there

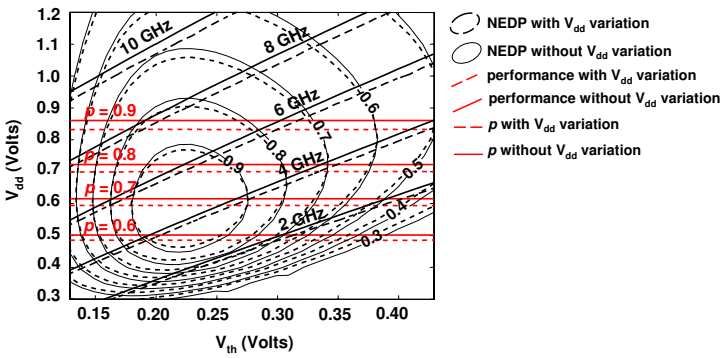


Figure 57: Constant NEDP and performance contours for a PCMOS full adder with V_{dd} variations when mean value plus one standard deviation is considered for EDP, performance, and p.

is no V_{th} variation.

7.4.3 Variations in Supply Voltage

In integrated circuits, effects such as IR voltage drop and SSN cause variations in the voltage level of the power supply [128]. Supply voltage variation affects both performance, energy consumption, and probability of correctness of the circuit. In this section, we assume that the variation in power supply is normally distributed with $3\sigma_{Vdd}$ equal to 10% of the mean value of the supply voltage [27]. Here, σ_{Vdd} is the standard deviation of the normal distribution.

In Figure 57, we illustrate the effect of the variation in V_{dd}. In the figure, the solid contours represent the results when there is no variation and the dashed contours correspond to the results when there is variation in V_{dd}. We consider a worst-case scenario, and hence show the contours corresponding to the mean plus one standard deviation in EDP, performance, and p. As seen from the figure, NEDP contours for the case when there is V_{dd} variation are narrower than and encircled by the NEDP contours for the case when there is no variation in V_{dd}. This is caused by the increased values of EDP, which is due to our choice of mean plus one standard deviation for EDP. Since we consider mean plus one standard deviation for performance and probability, their values are also increased. As a result, performance contours are shifted to the left and probability contours are shifted down.

Referring to the case when there is no variation in V_{dd} as the ideal case, our results have shown that EDP values when there is variation in V_{dd} differ from the EDP values of the ideal case by 5% on the average. Meanwhile, the difference between the performance values when there is variation in V_{dd} and the performance values of the ideal case is 7.2% on the average. Hence, V_{dd} variation has a higher impact on the performance. As V_{dd} increases, performance increases, but the energy consumption decreases. As a result, the overall impact of the V_{dd} variation on EDP is not as high as its impact on performance. Furthermore, V_{dd} variation has a slight impact on p. The difference between the p values when there is no variation in V_{dd} and the p values of the ideal case is 1.3% on the average.

Furthermore, the optimal values of V_{dd} and V_{th} change slightly. For example, when we consider minimizing the EDP of a full adder with the probability constraint $0.80 \leq p \leq 0.90$, and performance constraint $f \geq 6$ GHz, the optimal values of V_{dd} and V_{th} are now 0.70 V and 0.22 V with negligible difference compared to the optimal values of 0.72 V and 0.22 V obtained when there is no V_{dd} variation.

We can conclude from the results of this section that EDP and performance are dependent on the variations in the T, V_{th}, and V_{dd}. Similarly, our V_{dd} and V_{th} values for optimal EDP and optimal p points are also dependent on these variations. Thus, an analysis for optimizing the EDP of PCMOS devices and circuits should consider the variations in T, V_{th}, and V_{dd}. Furthermore, threshold voltage control and supply voltage control are necessary when the variations in T, V_{th}, and V_{dd} can not be modeled accurately.

7.5 Caveats on Energy and Probability Modeling

In this section, we will discuss two issues that we have not studied in the former sections. These issues are: (1) effect of noise on the energy consumption of PCMOS gates, and (2) effect of the filtering of noise on the probability parameter of a PCMOS gate.

7.5.1 Effect of Noise on Energy Consumption

In the previous sections, we have considered the energy consumed by a PCMOS gate per one switching, and we regarded switching as the charge (discharge) of the output capacitance of the gate from 0 V to V_{dd} V (V_{dd} V to 0 V). However, when noise is coupled to the

evaluation nodes of a gate, the voltages at the evaluation nodes undergo many spurious variations due to noise. Due to these spurious invocations, the gate consumes additional energy. Although this subject is out of the scope of this work, we will briefly explain it. We note that modeling the impact of noise on energy consumption is a difficult task. To model the impact of random fluctuations at the evaluation nodes of the gate, one can use the methods of statistical timing analysis [95] and statistical power analysis [30].

Let us now consider a probabilistic CMOS gate and a deterministic CMOS gate, wherein the deterministic gate produces a correct output with a probability nearly equal to 1 (with a very small probability of error) and the probabilistic gate produces a correct result at the output with a probability p ($0.5 < p < 1$). The probabilistic gate might consume more energy than its deterministic counterpart in a time interval, T, during which the deterministic gate undergoes switchings due to changes at its input voltage while probabilistic gate undergoes switchings due to the changes at its input as well as the spurious switchings due to noise. We inquired into the effect of the noise on the energy consumption of a PCMOS inverter (see [103]) and observed that the resulting growth in energy due to noise is negligible for small values of noise rms, but becomes significant as the noise rms value increases. Based on this observation and the results of circuit simulations, in our current work, we have considered noise rms values up to 0.2 V.

7.5.2 Effect of Noise Filtering on p

In this work, we have considered that noise is fully propagated through the path from the input to the output of a PCMOS circuits. Thus, in computing p, we used the rms value of the noise that is coupled to the evaluation nodes. However, if the duration of the noise pulse is shorter than the propagation delay of the gate, then not all the noise power is propagated from the input to the output of the gate since the high frequency components of the noise are filtered by the gate. We have studied this phenomenon in Chapter 3 through HSPICE simulations for a PCMOS inverter, and observed that the ratio of the noise duration to the propagation delay of the inverter affects the p of the gate such that when the noise duration is smaller compared to the propagation delay, the p value measured through simulations is

greater than the analytically calculated p. Thus, in circuit simulations of this work, we use noise durations greater than the propagation delay of the PCMOS gates that we consider.

We note that estimating the output of a PCMOS gate that is coupled with noise at its evaluation nodes involves an analysis of the nonlinear large signal behavior of the gate, and is beyond the scope of this work.

7.6 Conclusions

In this chapter, we have shown the design trade-offs between energy, performance, and p of PCMOS gates using analytical models of energy, propagation delay, and p. We have considered V_{dd} and V_{th} as our design variables and found the values of V_{dd} and V_{th} for optimal EDP and p under given constraints on p, performance, and EDP. We have observed that operating a PCMOS gate at lower supply voltages is more preferable to minimize its EDP. By contrast, for maximizing the p of a PCMOS gate, operating the gate at higher supply voltages is preferable. We have observed that the optimal values of V_{dd} and V_{th} are contingent upon the constraints imposed by the application as well as the models used for energy, delay, and p. We have also performed circuit simulations to validate our analytical models. From the simulations we have observed that the shapes of EDP surfaces and location of the optimal EDP point are dependent on the models used for energy, delay, and p. Our analysis can be helpful in circuit design for applications with specific p, performance, and EDP requirements. Two categories of candidate applications are the probabilistic applications and the error-tolerant applications will be described in Chapter 8.

We have also included an analysis of the impact of the variations in threshold voltage, temperature, and supply voltage on EDP, performance, and p of PCMOS gates as well as on optimal values of EDP and p. We have found that accurately estimating the variations in temperature, threshold voltage, and supply voltage is important for accurately optimizing the EDP and p of PCMOS gates.

Furthermore, we briefly discussed the effect of the noise on the energy consumption of a PCMOS gate, as well as the effect of the noise duration on the p of a PCMOS gate. Noise causing spurious switchings may increase the energy consumed by a PCMOS gate. The ratio

of the noise duration to the propagation delay of the gate is important in determining the p of a PCMOS gate. If the noise duration is smaller than the propagation delay of the gate, then high frequency components of the noise are filtered by the gate. Hence, we conclude that probability and energy models of probabilistic gates should consider these issues to accomplish accurate optimizations.

CHAPTER VIII

REALIZING ENERGY EFFICIENT ARCHITECTURES USING PCMOS TECHNOLOGY

To harness PCMOS technology to implement applications, two categories of applications have been considered: (i) applications which benefit from (or harness) probabilistic behavior at the device level, (ii) applications that can tolerate probabilistic behavior at the device level. In the context of the first category of applications, various applications from the cognitive and embedded domain which embody probabilistic behaviors were investigated [29]. These algorithms are probabilistic, that is, upon repeated execution with the same inputs, each step of these algorithms could have several possible outcomes, where each outcome is associated with a probability parameter. Examples of such algorithms include the celebrated probabilistic tests for primality [196]. Moving away from applications that embody probabilistic behaviors naturally (and in turn harness probabilistic behavior of PCMOS devices), the domain of applications that tolerate probabilistic behavior [55] is also considered in PCMOS research. Specifically, the applications which can trade energy and performance for application-level quality of the solution are investigated. Applications in the domain of digital signal processing are good candidates, where application-level quality of solution is naturally expressed in the form of signal-to-noise ratio or SNR.

In this chapter, we will outline the implementation approach for these applications. We will start with a description of the probabilistic system on a chip (PSOC) architectures which serve as implementation platforms for probabilistic applications. Following this, we will summarize the suite of probabilistic applications that have been considered. Then, we will summarize our approach and results for the signal processing applications.

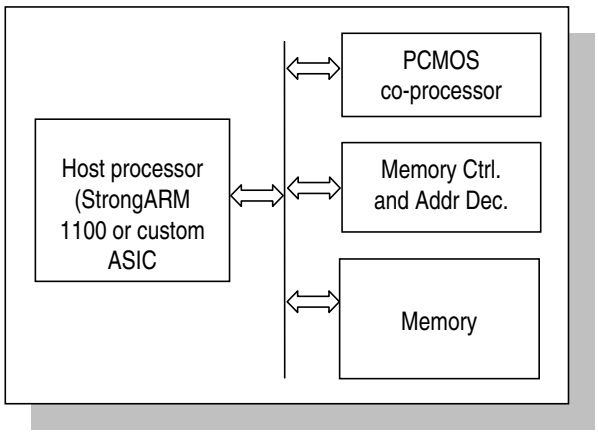


Figure 58: A probabilistic system-on-a-chip architecture.

*8.1 Probabilistic System on a Chip (*PSOC*) Architectures*

To realize energy efficient embedded computing platforms, we have developed a methodology for using PCMOS. As shown in Figure 58 (and discussed in detail by Chakrapani *et al.* [29]), a PSOC architecture is comprised of a host processor and a co-processor where the host processor is used to compute most of the control-intensive deterministic components of an application, whereas the co-processor realized using PCMOS can be viewed as an energy-performance accelerator that executes the probabilistic content of the application. A typical host processor will be a low-energy StrongARM [202], a MIPS [194] or an equivalent low-energy embedded processor, coupled to the co-processor via the system bus. Thus, the host processor accesses the co-processor through memory mapped I/O. The host could also be a custom-fit processor in its own right; for details about the concept of a SOC and further details about custom-fit processors, please see Lyonnard et al. [112], Tensilica [233] and Wang et al. [229].

The two basic criteria of interest in realizing efficient application-specific SOC architectures are the performance (typically the running-time of the application) and energy consumption (or its derivative power). Thus, our goal shall be to realize architectures that are significantly more efficient than a conventional processor using both of these criteria. Thus, our first *metric* for consideration shall be the *energy-performance* product of an architecture of a particular application—akin to the energy-delay product in circuit design—defined

below.

Energy-Performance Product (EPP). EPP is defined as the product of the application level energy (measured in Joules) and performance (measured in number of cycles).

EPP will be used as the chief metric of interest to evaluate various implementations. Given the EPP of two alternate implementations—for example, the case when the entire algorithm is implemented as software executing on the host referred to as the *baseline*, compared to the case where the deterministic part of the algorithm is executed on the host with the probabilistic part executing on a PCMOS co-processor—they can be compared by computing the ratio of their individual EPP values.

Since our goal is to compare the energy and performance gains realized through using PCMOS technology, we refine this notion and define the metric: EPP *gain*, which is denoted as Γ, and defined as follows.

EPP **Gain (Γ).** EPP gain, denoted as Γ, is the ratio of the EPP of the baseline to the EPP of a particular implementation. The EPP gain of a particular implementation $\mathcal{I}$ is determined as

$$\Gamma_{\mathcal{I}} = \frac{Energy_B \times Time_B}{Energy_{\mathcal{I}} \times Time_{\mathcal{I}}} \tag{112}$$

In Equation(112), unless otherwise stated *baseline* denoted as B refers to the case when the entire application is realized using software on the host (for example, a StrongARM SA-1100 processor or equivalently, an ASIC realization of deterministic content) only, *without recourse to a co-processor*. For example, in the context of a Bayesian network application, the well-known *Junction Tree* (deterministic) algorithm [114] shall serve as the baseline, and StrongARM SA-1100 shall be the baseline processor computing the deterministic as well as the probabilistic content, whereas $\mathcal{I}$ is the combination of the StrongARM SA-1100 as the host computing the deterministic component and the co-processor computing the probabilistic components of the application.

Alternate scenarios or "mixes" of partitioning the application across the host and co-processor are shown in Figure 59 as cases (b), (c) and (d). In the scenario wherein the target

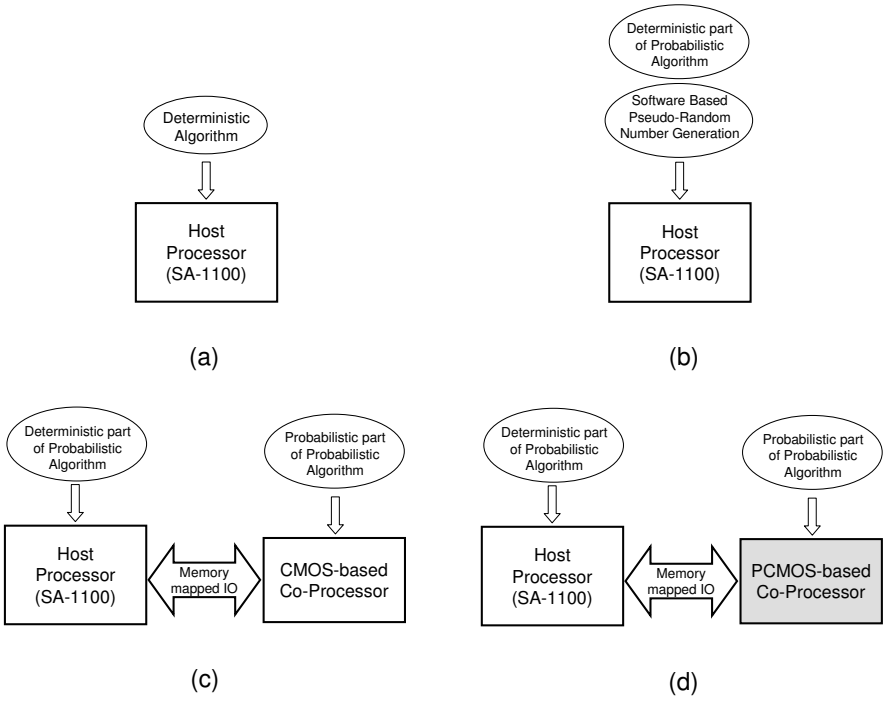


Figure 59: The four possible realizations of an application using an SOC platform wherein (a) a deterministic application is executed on the host only, (b) a probabilistic application is executed on the host only using an emulation based on pseudo-random bits generated using software, (c) the pseudo-random emulation is realized using a custom CMOS co-processor, and (d) a PCMOS co-processor is used to realize the probabilistic components of the application.

computational platform is constituted exclusively of a host without any co-processor—our baseline case—, and when considering a probabilistic variant of the Bayesian network for example, the probabilistic component is "emulated" using pseudo-random bit generation in software (as shown in Figure 59(b)).

8.2 The Suite of Applications

To demonstrate the utility and efficacy of a PSOC we consider applications that employ probabilistic algorithms that implement *bayesian networks* (BN) [178, 161], *random neural networks* (RNN) [54], *probabilistic cellular automata* (PCA) [53] and *hyper-encryption* (HE) [45]. Probabilistic content and utility in a wide range of application scenarios are the common characteristics of these algorithms.

Among these algorithm kernels that we have implemented, the Bayesian networks are

used in applications such as spam-filters, cognitive learning, battlefield planning [165], windows printer trouble shooting and hospital patient management [15]. Random neural networks are used in image and pattern classification, network routing and optimization of NP-hard problems such as finding the vertex cover of a graph. Probabilistic cellular automata are used in pattern or string classification [53], and finally, hyper-encryption is used in security applications for message encryption.

8.2.1 Example: Mapping the Bayesian Inference Algorithm to a PSOC Architecture

Bayesian inference is a statistical inference technique mimicking the human decision making process. *Hypotheses* and their corresponding *probability weights* are notions central to this technique. The probability weights are interpreted to be the degrees of belief associated with the corresponding hypotheses. Based on evidences, the degree of belief in a hypothesis is incremented (or decremented) till it approaches 1 (or 0) in which case the hypothesis is very likely (unlikely). A Bayesian network is used to perform a task referred to widely as Bayesian inference, and is modeled as a directed acyclic graph G of nodes V representing variables and edges E representing dependence relations between the variables. Each variable u uniquely represented by a node $v \in V$ can be assigned a value from a finite set of values $\sum_u$. Each value $\rho \in \sum_u$ has a conditional probability $p\left(\rho/\rho' \in \sum'\right)$ associated with it, where $\sum' \in (\sum_1 \times \sum_2 \times \sum_3 \cdots \sum_l)$ is the string of values of the variables represented by all the l parents of u. Variables whose values are known apriori are called *evidences* and based on such evidence, other variables are inferred. The particular Bayesian networks considered in this study is a part of the following applications: a hospital patient management system and printer troubleshooting in a Windows operating system environment.

The likelihood weighting algorithm [165] was chosen for Bayesian inference. The random experiment (used for inference) in this probabilistic algorithm, is implemented in the PCMOS co-processor (consisting of several modules), with the remainder implemented as software executing on the host. In a Bayesian network G, the conditional probabilities associated with each value of the variables of a node are known apriori, and are used to design a module of PCMOS switches (inverters), one module per node v in the graph. As

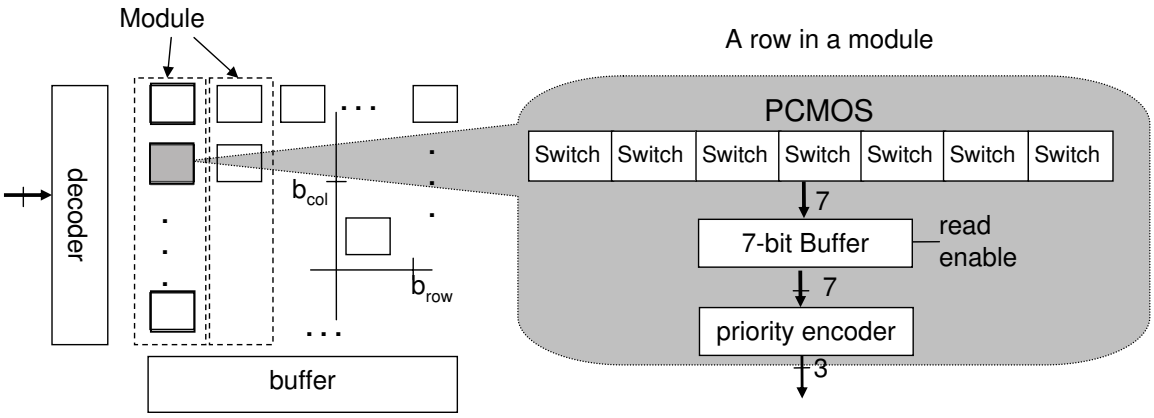


Figure 60: The co-processor architecture of a PSOC which implements Bayesian inference.

an example, consider a node u with $\sum_u = \{0, 1, 2\}$ and let $\sum'$ be an instance of the string of values associated with the parents of u. Let $0 \leq p\left(0/\rho'\right), p\left(1/\rho'\right), p\left(2/\rho'\right) \leq 1$ be the conditional probabilities associated with $0, 1, 2 \in \sum_u$ respectively, given that the parents of the node v have outputs $\rho' \in \sum'$. In our PSOC architecture, Bayesian inference will be performed by three PCMOS switches A,B and C corresponding to 0,1, 2 respectively. The inputs for these switches are fixed at 0 and the probability of correctness associated with A,B,C is by design, $p\left(0/\rho'\right)$, $\frac{p(1/\rho')}{1-p(0/\rho')}$ and $\frac{p(2/\rho')}{1-p(0/\rho')-p(1/\rho')}$ respectively. Thus, when the switches are inspected in the order $< A, B, C >$, the value which corresponds to the first switch whose output is the value 1 is the value inferred by node u. In the PSOC design, the set of switches $\{A, B, C\}$ will be referred to as a row and each distinct switch in this set will be referred to as an *element*. Since a row is associated with each element of the set $\sum'$, many rows are required to implement the strings associated with the space of all possible outputs corresponding to the parents of the node u from $\sum'$. These set of rows will be referred to as a *table*.

As shown in Figure 60, the PCMOS module corresponding to a node u implements a table, whose row is indexed by a particular string ρ' of values associated with the parents of u computed earlier. The number of columns in the table is $|\sum_u|$, where each column corresponds to a value from the set $\sum_u$; in our example, $|\sum_u| = 3$. An element in the table, identified by $<$ row, column $>$ is a specialized PCMOS switch whose probability of correctness is computed as indicated above. Finally a conventional priority encoder is connected

to the outputs of a row to determine the final result of the random experiment; it performs the function of inspecting the values of a row and choosing the final output associated with u. Note that, each PCMOS switch also includes an amplifier in it, that is, each PCMOS switch corresponds to a RESINA element. Hence, in computing the application level gains that will be reported next in Section 8.4, the energy cost of amplification is also considered.

8.3 Applications that Tolerate Probabilistic Behavior

In this section, we will consider the domain of applications that tolerate probabilistic behavior. Specifically, we investigate applications which can trade energy and performance for application-level quality of the solution. Applications in the domain of digital signal processing are good candidates, where application-level quality of solution is naturally expressed in the form of signal-to-noise ratio or SNR.

To demonstrate the utility of PCMOS technology in this context, filter primitives that are using PCMOS technology are used to realize the H.264 decoding [231] and synthetic aperture radar (SAR) [179] imaging algorithms. In particular, PCMOS devices are used to build probabilistic arithmetic elements, such as adders and multipliers, to realize energy efficient computing elements that can be used to perform DSP primitives such as FFT and FIR filter. In these computing elements, the probabilistic behaviors are induced by lowering the supply voltage V_{dd} in the presence of noise affecting them. To achieve application level quality, which, for example, can be the signal to noise ratio (SNR) in the case of an FIR filter, V_{dd} is scaled in a biased manner [55]. In this biased voltage scaling (BIVOS) scheme, the V_{dd} of the circuitry computing the most significant bits is scaled less than the V_{dd} of the circuitry computing the less significant bits. Thus, the most significant bits yield a higher probability of correctness, which in turn improves the probability of correctness of the corresponding DSP application.

Using the PSOC framework sketched before in this case, the application is partitioned in a manner where the core control-flow such as branches will be executed on the host processor whereas the signal processing kernels will be executed on the probabilistic co-processor. For example, for synthetic aperture radar (SAR) imaging application, the co-processor realizes

the FFT primitive.

8.4 Application Level Gains of PCMOS

For probabilistic applications, as summarized in Table 10, gains at the scope of an entire application range from a factor of about 80 for the PCA application, to a factor of about 300 for the RNN application. Here, the baseline implementation for HE, PCA and RNN applications is the StrongARM SA-1100 computing the deterministic as well as the probabilistic content and $\mathcal{I}$ is a PSOC executing an identical probabilistic algorithm. For the BN case, the baseline is the StrongARM SA1100 computing the deterministic junction tree algorithm and $\mathcal{I}$ is a PSOC executing the likelihood weighting algorithm. A range of EPP gains are observed whenever multiple data points are available, for example, in the context of the Bayesian inference where different data points correspond to different networks, gain varies from a factor of 12.5 to an impressive factor of 291. Note that this increase in the gain is due to the increase in the flux, which is defined as the *ratio of probabilistic operations to the total number of operations* of the algorithms.

Table 10: Application level min and max EPP gains of PCMOS over the baseline implementation, where the StrongARM SA-1100 processor serves as the host.

Application	Baseline	EPP Gain of PCMOS = Γ_{PCMOS}	
		Min	Max
BN	Deterministic junction-tree alg. on StrongARM (Figure 59(a))	12.5	291
RNN	Probabilistic alg. on StrongARM (Figure 59(b))	226.5	300
PCA		61	82
HE		1.12	1.12

One observation is that—besides probabilistic content—the EPP gain also depends on the efficiency of the host serving as the baseline. If the energy consumed on the host to compute the deterministic part of an application is more dominant than the energy consumed to compute the probabilistic part on the co-processor, then the EPP gain would be very small. This fact is observed in the case of the HE application, where the host SA-1100 energy is dominant and hence the resulting EPP ratio is only 1.12 (see Table 10). When the StrongARM host is replaced by its much more efficient ASIC counterpart, the

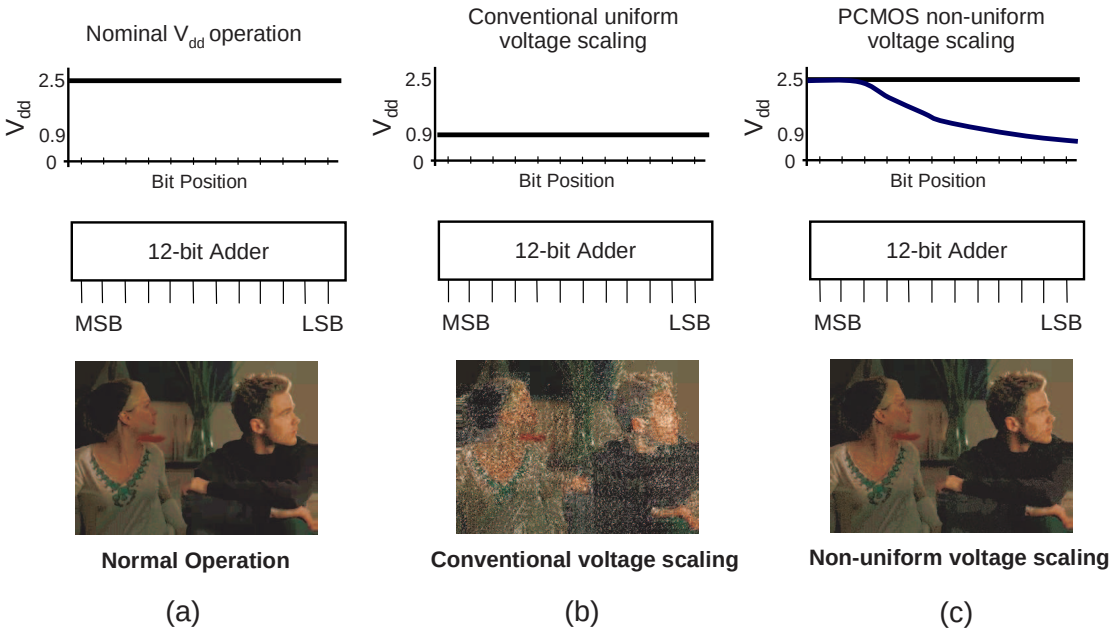


Figure 61: Comparing images reconstructed using H.264 (a) conventional error free operation (b)probability parameter p lowered uniformly for all bits (c) probability parameter p varied non-uniformly based on bit significance.

gains increase significantly, from 1.06 to 9.38 in the case of hyper-encryption and from 82 to 561 in the case of the probabilistic cellular automata.

A set of results for the DSP applications are shown in Figure 61. As illustrated in Figure 61(b) the probability parameter p of correctness can be lowered uniformly for each bit in the adder (which is one of the building blocks of the FIR filter used in the H.264 application). While this approach saves energy, it significantly degrades the output picture quality when compared to conventional (CMOS based and error-free) operation. However, as illustrated in Figure 61(c), if the probability parameter is varied non-uniformly, significantly lower energy consumption is possible with minimal degradation of the quality of the image [55, 156]. Hence, not only can PCMOS technology be leveraged for implementing energy efficient filters, but also can be utilized to naturally trade-off energy consumed for application level quality of solution through novel probabilistic voltage scaling schemes [55, 156]. Additionally, the efficiency of this approach is also examined through the metric of energy (measured in Joules) performance (measured in seconds) product (EPP), as well as the EPP per dB gain. The results are summarized in Table 11: non-uniform voltage scaling is far

Table 11: SAR Performance.

Voltage Scaling Scheme	SNR	Energy	Running Time	EPP	EPP/ SNR
BIVOS	28dB	1/5.6X	2.5X	0.44X	15.7×10^{-3}
Uniform Voltage Scaling	0dB	1/2.5X	1.41X	0.56X	∞

less expensive in terms of the EPP cost per dB, denoted EPP / SNR. Shown in Table 11, non-uniform voltage scaling yields an EPP / SNR expense of 15.7×10^{-3}, compared to ∞ in the case of uniform voltage scaling.

8.5 Conclusions

In this chapter, we have demonstrated the value of the novel PCMOS technology within the context of realizing ultra efficient PSOC architectures over a range of applications. The improvements that were demonstrated for probabilistic applications were orders of magnitude over application specific CMOS designs. For DSP applications, the concept of probabilistic arithmetic was introduced and shown to be effective in realizing energy efficient signal processing. This led to the novel BIVOS approach for designing PCMOS based probabilistic arithmetic primitives. The details of the energy efficient PSOC platforms for probabilistic and DSP applications can be found in [29] and [55], respectively.

CHAPTER IX

FUTURE DIRECTIONS FOR PCMOS RESEARCH

In the former chapters, we have addressed the issue of probabilistic design at several levels from circuits to various applications from the cognitive and embedded domains with an emphasis on PCMOS circuits. In this chapter, we will discuss the future opportunities for PCMOS research. We will start with explaining the possible extensions to PCMOS research with a focus on the probabilistic circuit and system design. We will also briefly discuss the research opportunities in the domain of applications. Following this, we will consider the applicability of probabilistic computing to emerging technologies. Furthermore, we will compare our PCMOS switch to a single electron over-barrier transport based switch. Finally, we will come back to CMOS technology and summarize our projections for the characteristics of PCMOS circuits to future semiconductor technologies.

9.1 Future Directions and Challenges for PCMOS Research

This dissertation research has provided an extensive characterization of PCMOS circuits in terms of their energy consumption, probability of correctness and performance by considering various issues such as the different ways that noise can be coupled to the circuit, the effects of noise and output sampling frequencies and the effects of short-circuit and subthreshold leakage currents. However, the issue of gate leakage is not discussed. Considering that gate leakage has gained significant importance due to the increasing proportion of it to the total leakage current with more advanced technologies, an important future consideration for PCMOS research is the study of the impact of the gate leakage on the energy-probability relationship of PCMOS circuits.

Another interesting problem for PCMOS research is the synthesis of probabilistic and error-tolerant circuits, that is, the synthesis of circuits from gates with probabilistic behavior. Using the foundations introduced by this dissertation to analyze PCMOS circuits, it

would be possible to develop synthesis methodologies for probabilistic and error-tolerant circuits.

Another future research goal is to extend the notions of (noise-induced) time-varying devices—which are extensively characterized in this dissertation, to their (variation-induced) space-varying counterparts through the concept of statistical stationarity. Statistical stationarity allows a random process' moments such as the mean or variance (averaged) over time, to be equated with those (averaged) over an ensemble. In our case, the ensemble will represent the distribution of a parameter (e.g., V_{th}) over the set of devices scattered on surface of silicon whereas the probabilistic behavior induced by noise is a time varying phenomenon. Thus, statistical stationarity can be used to extend the techniques that we have already validated for coping with noise-driven probabilistic devices to the context of the parameter variation-tolerance.

In the context of characterizing parameter variations, different variation sources such as die-to-die variations and within-die variations can be studied. In the context of probabilistic applications, one can explore ways on how to treat these variations as sources of randomness, whereas in the context of error-tolerant (signal processing) and deterministic applications it is possible to explore ways on how to apply error correction/recovery/redundancy mechanisms for these different variation sources.

9.1.1 Implementation Challenges

The actual implementation and fabrication of architectures that leverage PCMOS based devices poses further challenges. Chief among them is "tuning" the PCMOS devices, or in other words, controlling the probability parameter p of correctness. Additionally, the number of distinct probability parameters is a concern, since this number directly relates to the number of voltage levels. In the circuit level, we need (1) to analyze the dependency of p to process variations (2) to investigate the methods that will let us tune the p values adaptively. In the application level, more specifically, for the probabilistic applications we make two observations aimed at addressing these problems: (i) The distinct probability parameters are a requirement of the application and the *application sensitivity* to probability

parameters is an important aspect. That is, if an application uses probability parameters p_1, p_2, p_3, for example, it might be the case that the application level quality is not affected when only two distinct values, say p_1, p_2 are used. This, however can only be determined experimentally and is a topic being investigated. (ii) Given probability parameters p_1 and p_2, other probability parameters might be derived through logical operations. For example, if the probability of obtaining a 1 from a given PCMOS device is p and the probability of obtaining a 1 from a second PCMOS device is q, a logical AND of the output of the two PCMOS devices produces a 1 with a probability $p.q$. Using this technique, in the context of an application, the number of distinct probability parameters may be drastically reduced. Since the probability parameter p is controlled through varying the voltage, reducing the number of probability parameters reduces the number of distinct voltage levels required and is another topic being investigated. Furthermore, the energy and performance cost associated with the additional voltage generation and supply routing necessary to provide multiple supply voltages has to be considered in the future.

9.1.2 Future Directions in the Domain of Applications

One important aspect of our study in extending our existing methodology in the domain of applications will be on statistics of the input signals. Since the device characteristics stay invariant over time, the resulting errors will depend on the input data that change over time. Errors due to delay variations, for example, occur only for those input data combinations that require propagation through the critical paths in the circuit. In this context, our aim is to identify and investigate the effects of input data sets on the probabilistic characterization of building blocks (e.g., adder, multiplier, and others) and evaluate their behavior when they are used in different applications. For example, these building blocks can be used as part of different computing units from those requiring deterministic operation (e.g., an ALU in a general purpose processor) to those that can tolerate errors (e.g., a signal processing primitives such as FIR and FFT). In the former case, an accompanying error correction/recovery mechanism will be needed, whereas in the latter, the error correction/recovery mechanism can be used so as to tune the degree of error depending on

the application requirements.

Thus, it is anticipated that a lot of research effort will be invested in designing an error correcting and recovery circuit, specifically due to deterministic applications that cannot tolerate errors as well as to error-tolerant applications to control the application quality.

9.2 Future Implementations of Probabilistic Switches

As described in Chapter 3, a probabilistic switch is a device which can be turned on or off with a concomitant energy expenditure. The on and off states are detected by an act called an observation which corresponds to a measurement of the voltage or current, and the outcome of an energy expending action to turn the switch on or off is randomized in that it will occur with a fixed a priori probability p. In realizing this switch using PCMOS approach, we leverage the probabilistic behaviors, in particular the noise available in CMOS circuits. In the heart of our approach is the idea of exploiting the noise, which is conventionally an undesired phenomena. Inspired by this, in this section, we will consider two of the emerging research technologies, single electron transistor (SET)s and carbon nanotube (CNT)s, and discuss the implementation opportunities for probabilistic switches using these technologies. We will also consider an over-barrier transport based switch and compare its probabilistic behavior with our PCMOS switch.

9.2.1 Realizing Probabilistic Switches with SETs

A probabilistic switch can be implemented using SETs as shown in Figure 62. The device shown in Figure 62 is a single-electron switch with differential inputs (Asahi [7]). In this switch, there are four tunnel junctions (J_1 through J_4) and three capacitances (C_1 through C_3). The input voltage X is applied through capacitor C_2 and complement of X is applied through capacitor C_3. If X is an appropriate positive voltage, then the input value is interpreted as binary 1. On the other hand, if X is a proper negative voltage then the input value is binary 0. Tunneling through junctions J_1 and J_3 is controlled by the charge on the capacitors C_1, C_2 and C_3, the values of X and $\bar{X}$, and the value of clock (ϕ) voltage. For example, an electron tunnels through J_1 towards B if the charge on the right side of J_1 is more positive than the charge on the left side of J_1, wherein the charge on the right side is

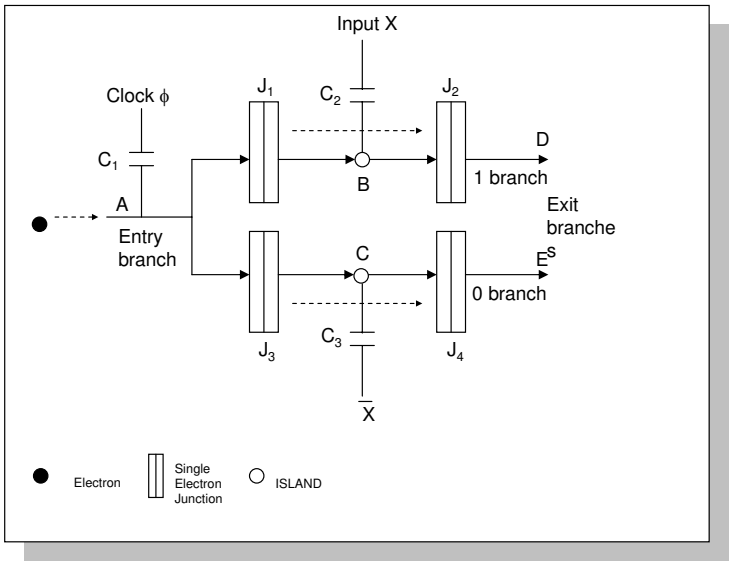


Figure 62: SET-based implementation of a probabilistic switch.

dependent on the capacitance of C_2, the value of X, and the value of ϕ; and similarly the charge on the left side is dependent on the capacitance of C_2, the value of $\bar{X}$, and the value of ϕ. When an electron is supplied at the entry branch, it follows the path $A \rightarrow B \rightarrow D$ (the 1 branch) when X has a positive value, and $A \rightarrow C \rightarrow E$ (the 0 branch) if X has a negative value. More specifically, if the electron follows the path $A \rightarrow B \rightarrow D$,this is interpreted as a switching to 1; whereas if it follows the path $A \rightarrow C \rightarrow E$, this is interpreted as a switching to 0.

Tunneling is a probabilistic phenomenon, and the waiting time for the expected tunneling is not fixed. Hence, the duration of the clock signal is critical. For the device operation without error, the clock duration should be sufficiently long. The probability of that the waiting for the expected tunneling will be longer than the clock period (t_{CLK}) is given by

$$p_e = exp\,(-t_{CLK}\Gamma) \tag{113}$$

where Γ is the mean tunneling rate. p_e is the probability of incorrect operation and $p = 1-p_e$ is the probability of being correct for this implementation of a probabilistic switch.

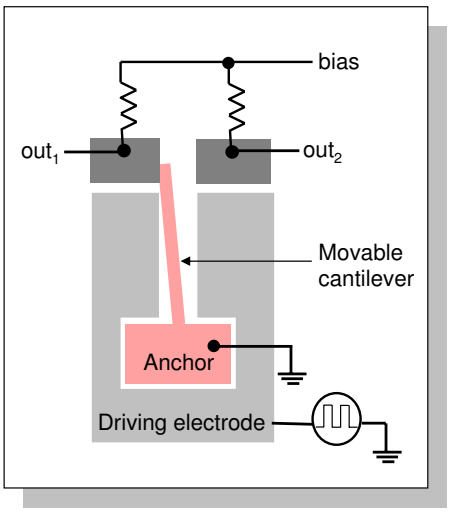


Figure 63: CNT-based implementation of a probabilistic switch.

9.2.2 Realizing Probabilistic Switches with CNTs

Carbon nanotubes are molecular-scale tubes of graphitic carbon with outstanding properties. They are among the stiffest and strongest fibers known, and have remarkable electrical properties. They can be used to design nanoscale sensors, actuators, devices and systems (collectively referred to as Nanoelectromechanical Systems (NEMS). NEMS applications include the random access memory [209], nanotweezers [98] and electrostatic switches [12]. Even though NEMS can be designed using a number of materials, carbon nanotube based NEMS are attractive as, for example, carbon nanotube based electrostatic switches have the potential to offer extremely high resonant frequencies in the gigahertz range because of their high stiffness.

Figure 63 illustrates the concept of a CNT based probabilistic switch. Here, a movable cantilever is placed in the middle of a U-shaped electrode. The cantilever is actuated by the electrodes places on either side, and thermal fluctuations and mechanical noise cause the cantilever to randomly contact one of the output electrodes. Hence, random strings of "1"s and "0"s are measured at the outputs.

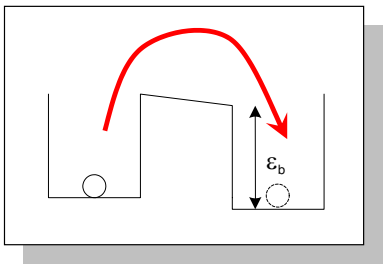


Figure 64: An over-barrier transport based binary switch.

9.2.3 Comparison of the PCMOS Switch to an Over-Barrier Transport Based Binary Switch

Having discussed two practical examples of implementing probabilistic switches using emerging research technologies, in this section, we will consider a fundamental binary switch and present a brief theoretical discussion of the differences between the two switches. The reason for considering the over-barrier transport based binary switch is that it is the most fundamental computational element which has been used to derive theoretical limits on device scaling [124, 236].

Figure 64 shows an over-barrier transport based binary switch, which is a single electron device consisting of two wells separated by a finite potential barrier. This switch has two stable states. The location of the electron, or the state of the switch can be changed either by supplying the electron additional energy or, equivalently, by reducing the barrier energy, denoted by ϵ_b. When the two wells are separated by the potential barrier, ideally electron should not move between the wells. However, due to the spontaneous transitions, the electron can move between the wells leading to a probability of error p_e. The location of the electron is said to be *distinguishable* if there is a very low probability of electron spontaneously moving to the other well. If, on the other hand, for the election in a given state (well), the probability of spontaneous transition to the alternate state (well) is equal to 0.5, then distinguishability is lost. A spontaneous transition can occur either due to classical over-barrier or quantum mechanical tunneling transitions. In this section, we assume that the barrier width is large enough and only over-barrier transitions are taking place. From

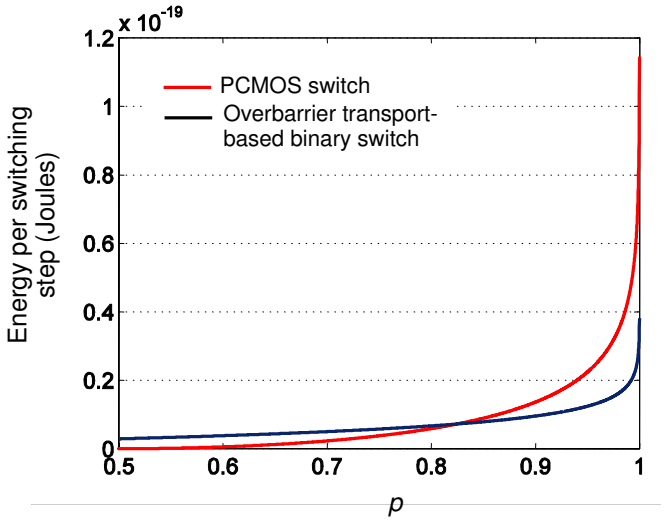


Figure 65: Comparison of the PCMOS switch with the over-barrier transport based binary switch.

Boltzmann probability distribution, the probability of such a classical transition is

$$p_e = exp\left(-\frac{\epsilon_b}{kT}\right) \tag{114}$$

where k is the Boltzmann's constant and T is the temperature. When (114) is solved for $p_e = 0.5$, that is, the case when distinguishability is completely lost, the well-known energy limit per switching operation, $\epsilon_b = kTln2$, is found. The probability of this switch not making spontaneous erroneous transitions is $p = 1 - p_e$. Hence, the barrier height ϵ_b is related to p through

$$\epsilon_b = kTln\left(\frac{1}{1-p}\right) \tag{115}$$

Now, we will compare this fundamental switch to our PCMOS switch. To be able to make a fair comparison, we consider a PCMOS switch with only inherent thermal noise (kT/C noise [198]). The energy-probability relationship for such a PCMOS switch is

$$E = 4kT\left[inverf\left(2p-1\right)\right]^2 \tag{116}$$

In Figure 65, we compare the energy-probability relationship of these two switches. As seen from the figure, the two switches exhibit similar characteristics, for example for both of the switches, the rate of the increase in energy consumption with p increases as p converges to 1. However, the energy-probability characteristics of the switches are not exactly same

since the fundamental mechanisms governing the switches are different. Compared to the charge transport based switch, our PCMOS switch model is a macroscopic model and is not immediately amenable to analyzing the fine-grained limit estimates. Furthermore, the over-barrier transport based switch is characterized by Boltzmann distribution on non-negative energy levels, whereas PCMOS switch characterized by Gaussian distribution with mean 0.

One major difference between the switches is that when the probability value is 0.5, the energy consumed per switching of the PCMOS switch is 0, whereas it is $kTln2$ for the over-barrier transport based switch. This is due to the differences in interpretation of switching and its associated energy consumption for the two switches. For the PCMOS switch, the probability of error is associated with the switching, and each switching consumes energy E per switching. When $p = 0.5$, there is only noise in the circuit, hence noise can cause spontaneous switchings with no external energy investment. On the other hand, for the over-barrier transport based binary switch, probability of switching error is based on the distinguishability requirement. This switch switches correctly with $p = 1$ when the barrier height is reduced to zero, or the particle acquires energy ϵ_b. Hence, the amount of energy consumed for a correct switching is ϵ_b. For this switch, $\epsilon_b = kTln2$ denotes the barrier height that is required to guarantee distinguishability. When $p = 0.5$ and the barrier height is $\epsilon_b = kTln2$, then without any external energy investment this switch will experience spontaneous transitions. However, when the electron energy is increased by $kTln2$, or the barrier height is decreased to 0, corresponding to an energy consumption of $kTln2$, then this switch will switch correctly. Hence, we can say that when $p = 0.5$, similar to the PCMOS switch the over-barrier transport based binary switch can also make spontaneous transitions (hence switchings) without any investment of external energy.

9.3 PCMOS and Future Semiconductor Technologies

In this section, we will consider PCMOS from the perspective of future semiconductor technologies. We will first indicate the increasing importance of noise in semiconductor technologies and the necessity of probabilistic approaches to electronic design. Following this, we will discuss the utility of PCMOS for low energy consumption in future technologies.

From 2005 ITRS Roadmap

Year of Production	2006	2008	2010	2012	2014	2016	2018	2020
MPU Physical Gate Length (nm)	28	23	18	14	11	9	7	6
V_{dd} (low power) (V)	0.9	0.8	0.7	0.7	0.5	0.5	0.5	0.5
Equivalent oxide thickness (A) *				7	6	5	5	5
Total gate capacitance (AF) *				8.80	6.15	4.725	3.136	2.172

* Double gate MOSFET

Figure 66: Estimated values of physical gate length, supply voltage, oxide thickness and total gate capacitance from 2005 ITRS roadmap.

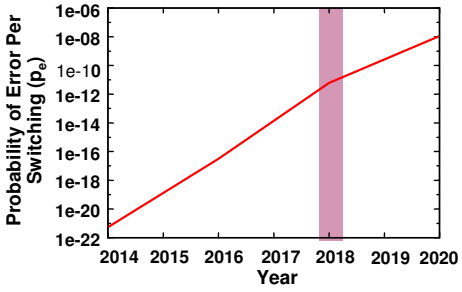


Figure 67: The variation of the probability of switching error due to the inherent noise across years.

In Figure 66, we list the values of physical gate length, V_{dd}, oxide thickness and total gate capacitance from year 2006 to 2020. These values are borrowed from the 2005 ITRS roadmap [82]. As seen from the figure, capacitance values are decreasing as time progresses. Hence, the inherent thermal noise (kT/C noise [198]) is growing due to the reduction in capacitance values with scaling. For example, in year 2018, gate capacitance is estimated to be 3.316×10^{-18} F, which corresponds to a noise voltage with rms value of 36.33 mV. From equation (39) in Section 3.2, the probability of error per switching will be 5.96×10^{-12} (also shown in Figure 67). Hence, a single device operating at 60 Ghz can produce 1300 errors in one hour, which is a very high error rate [100, 149]. Scaling beyond 2018 with such a big error rate is not plausible. On the other hand, to overcome the error, supply voltage values may be increased, but this would lead to an increase in energy consumption. Therefore, approaches which can cleverly trade-off energy and probabilistic behaviors are required, and PCMOS is one of these approaches. However, the utility of PCMOS into the future semiconductor technologies should be investigated more. For example, with the advancements

in semiconductor technology, gate leakage is becoming more significant. Considering that gate leakage energy does not scale with the supply voltage as fast as the switching energy, energy savings gained by PCMOS could be lower in the future. Furthermore, due to the limitations in the values of threshold voltage and supply voltage, the voltage scaling may be limited, and hence, energy savings by using PCMOS may also be limited.

9.4 Conclusions

In this chapter, we have discussed the future opportunities for PCMOS research. We have identified the possible extensions to the current research and the challenges that await PCMOS in the future CMOS technologies. We have also proposed new research directions. Furthermore, we have briefly discussed the implementation opportunities for probabilistic computing elements using the emerging research technologies.

CHAPTER X

CONCLUSIONS

Motivated by the necessity to consider probabilistic approaches to future designs, the main objective of this thesis was to develop and characterize energy efficient probabilistic CMOS circuits that can be used to implement low energy computing platforms. A probabilistic inverter (or switch) is the simplest gate that was considered. An extensive characterization of the behavior of the probabilistic inverter, based on a study of input- and output coupled thermal noise, as well as the effects of power supply noise was developed. Since the frequency at which the noise as well as the output voltage is sampled affects the probability of correctness, p, of the inverter, analytical models were developed to capture the effects of these two parameters. The relationship between the energy and probability of the probabilistic inverter that was established through analytical models and circuit simulations was also verified through physical measurements. A short-circuit energy model was developed to account for the effect of the short-circuit energy dissipation of the probabilistic inverter on its energy-probability relationship. The results of the characterization of the probabilistic inverter and the proposed analytical models can be used by circuit designers to realize energy efficient probabilistic designs.

The characterization of a probabilistic inverter was also extended to larger circuits. The probabilistic behavior of larger circuits was analyzed by first developing probability models of primitive gates, which are then input to a graph-based model to find the probabilities of larger circuits. The analysis of larger probabilistic circuits provides a basis for analyzing probabilistic behaviors due to noise in future technologies, and can be used in probabilistic design and synthesis methods to improve circuit reliability.

When the supply voltage of a PCMOS circuit is reduced, its p as well as its performance decreases. The performance loss can be compensated through decreasing the threshold voltage, but a reduction in the threshold voltage causes an increase in the leakage energy,

hence the total energy consumption of the circuit. These design trade-offs between energy, performance, and probability of PCMOS gates were studied. Based on this study, various methods were proposed to optimize EDP and p under given constraints on p, performance, and EDP. The impact of the variations in threshold voltage, temperature, and supply voltage on p, performance, and EDP, as well as the optimal values of EDP and p were also considered.

PCMOS circuits can be used to realize ultra efficient PSOC architectures over a range of applications. A thermal noise based RNG was developed to implement probabilistic applications using PCMOS technology. Th RNG, which included a subthreshold amplifier with very low energy consumption produced high quality random bits. For probabilistic applications, the energy and performance improvements gained by using PCMOS technology were orders of magnitude over application specific CMOS designs. For DSP applications, the concept of probabilistic arithmetic was shown to be effective in realizing energy efficient signal processing.

It was also established that there are many research areas that the ideas of PCMOS can be extended to and the PCMOS research will mature as the technology progresses.

REFERENCES

[1] Abbas, M., Ikeda, M., and Asada, K., "Noise immunity investigation of low power design schemes," *IEICE Trans. Electronics*, pp. 1238–1247, Aug. 2006.

[2] Abdollahi, A., Fallah, F., and Pedram, M., "Leakage current reduction in CMOS VLSI circuits by input vector control," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 12, pp. 140–154, Feb. 2004.

[3] Acar, E., Arunachalam, R., and Nassif, S. R., "Predicting short circuit power from timing models," in *Proc. Asia and South Pacific Design Automation Conf. (ASP-DAC)*, pp. 277–282, Jan. 2003.

[4] Agarwal, A. and Roy, K., "A noise tolerant cache design to reduce gate and sub-threshold leakage in the nanometer regime," in *Proc. Int. Symp. Low Power Electronics and Design*, pp. 18–21, Aug. 2003.

[5] Alvandpour, A., Larsson-Edefors, P., and Svensson, C., "Separation and extraction of short-circuit power in digital CMOS VLSI circuits," in *Proc. Int. Symp. On Low Power Electronics and Design*, pp. 245–249, Aug. 1998.

[6] Amusan, O. A., Witulski, A. F., Massengill, L. W., Bhuva, B. L., Fleming, P. R., Alles, M. L., Sternberg, A. L., Black, J. D., and Schrimpf, R. D., "Charge collection and charge sharing in an 130 nm CMOS technology," *IEEE Tran. Nuclear Science*, vol. 53, pp. 3253–3258, Dec. 2006.

[7] Asahi, N., Akazawa, M., and Amemiya, Y., "Single-electron logic device based on the binary decision diagram," *IEEE Tran. Electron Devices*, vol. 44, pp. 1109–1116, July 1997.

[8] Austin, B. L., Bowman, K. A., Tang, X., and Meindl, J. D., "A low power transregional MOSFET model for complete power-delay analysis of CMOS gigascale integration (GSI)," in *Proc. Annual IEEE Int. ASIC Conf.*, pp. 125–129, Sept. 1998.

[9] Bachtold, A., Hadley, P., Nakanishi, T., and Dekker, C., "Logic circuits based on carbon nanotubes," *Physica E*, vol. 16, pp. 42–46, 2003.

[10] Badaroglu, M., Tiri, K., der Plas, G. V., Wambacq, P., Verbauwhede, I., Donnay, S., Gielen, G. E., and Man, H. J. D., "Clock-skew-optimization methodology for substrate-noise reduction with supply-current folding," *IEEE Tran. Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, pp. 1146–1154, June 2006.

[11] Basu, A., Lin, S. C., Wason, V., Mehrotra, A., and Banerjee, K., "Simultaneous optimization of supply and threshold voltages for low-power and high-performance circuits in the leakage dominant era," in *Proc. Design Automation Conf. (DAC)*, pp. 884–887, 2004.

[12] Baughman, R. H., "*et al.*, carbon nanotube actuators," *Science*, vol. 284, pp. 1340–1344, May 1999.

[13] Bayes, T., "An essay toward solving a problem in the doctrine of chances," *Philosophical Transactions of the Royal Society of London*, vol. 53, pp. 370–418, 1764.

[14] Beckett, P. and Goldstein, S. C., "Why area might reduce power in nanoscale CMOS," in *Proc. IEEE Int. Symp. Circuits and Systems*, pp. 2329–2332, May 2005.

[15] Beinlich, I., Suermondt, G., Chavez, R., and Cooper, G., "The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks," in *Proceedings of the Second European Conf. on AI and Medicine*, pp. 247–256, 1989.

[16] Bennett, C. H., "Logical reversibility of computation," *IBM J. Research and Development*, vol. 17, pp. 525–532, Nov. 1973.

[17] Bernoulli, J., *Ars Conjectandi*. Basilea: Impenfis Thurnisiorum Fratrum, 1713. [A translated version: J. Bernoulli (author), E. D. Sylla (translator), *The art of Conjecturing, together with Letter to a Friend on Sets in Court Tennis*, Baltimore, Maryland: The Johns Hopkins University Press, Dec. 2005].

[18] Bhardwaj, S., Yu, C., and Vrudhula, S., "Statistical leakage minimization through joint selection of gate sizes, gate lengths and threshold voltage," in *Proc. Asia and South Pacific Conf. Design Automation*, pp. 953–958, Jan. 2006.

[19] Bhunia, S., Mukhopadhyay, S., and Roy, K., "Process variations and process-tolerant design," in *Proc. Int. Conf. VLSI Design*, pp. 699–704, Jan. 2007.

[20] Bisdounis, L. and Koufopavlou, O., "Short-circuit energy dissipation modeling for submicrometer CMOS gates," *IEEE Tran. Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, pp. 1350–1361, Sept. 2000.

[21] Boltzmann, L., *Lectures on Gas Theory*. New York: [translated by S.G. Brush], Dover Publications, 1965.

[22] Borel, E., *Leçons sur la théorie des fonctions*. Paris: Gauthier-Villars, 1898.

[23] Borkar, S., "Designing reliable systems from unreliable components: the challenges of transistor variability and degradation," *IEEE Micro*, vol. 25, pp. 10–16, Nov./ Dec. 2005.

[24] Borkar, S., Karnik, T., Narendra, S., Tschanz, J., Keshavarzi, A., and De, V., "Parameter variations and impact on circuits and microarchitecture," in *Proc. Design Automation Conf. (DAC)*, pp. 338–342, June 2003.

[25] Burnett, D., Higman, J., Hoefler, A., Li, C., and Kuhn, P., "Variation in natural threshold voltage of NVM circuits due to dopant fluctuations and its impact on reliability," *Int. Electron Devices Meeting*, pp. 529–532, Dec. 2002.

[26] Calhoun, B. H., Wang, A., and Chandrakasan, A., "Modeling and sizing for minimum energy operation in subthreshold circuits," *IEEE J. of Solid-State Circuits*, vol. 40, pp. 1778–1786, 2005.

[27] Cao, Y., Gupta, P., Kahng, A. B., Sylvester, D., and Yang, J., "Design sensitivities to variability: Extrapolations and assessments in nanometer VLSI," in *IEEE Int. ASIC/SOC Conf.*, pp. 411–415, Sept. 2002.

[28] Carnot, S., *Reflexions sur la Puissance Motrice du feu sur les machines propres a developer cette puissance.* Paris: Bachelier, 1824.[English translation: S. Carnot, *Reflections on the Motive of Fire*, Peter Smith Publisher, 1992].

[29] Chakrapani, L. N., Akgul, B. E. S., Cheemalavagu, S., Korkmaz, P., Palem, K. V., and Seshasayee, B., "Ultra-efficient (embedded) SOC architectures based on probabilistic CMOS (PCMOS) technology," in *Proc. Design Automation and Test in Europe (DATE)*, pp. 1110–1115, Mar. 2006.

[30] Chang, H. and Sapatnekar, S. S., "Full-chip analysis of leakage power under process variations, including spatial correlations," in *Proc. Design Automation Conf. (DAC) 2005*, pp. 523–528, June 2005.

[31] Chang, J., Abidi, A. A., and Viswanathan, C. R., "Flicker noise in CMOS transistors from subthreshold to strong inversion at various temperatures," *IEEE Trans. on Electron Devices*, vol. 41, pp. 1965–1971, Nov. 1994.

[32] Chang, T.-H., "Minimizing switching noise in a power distribution network using external coupled resistive termination," *IEEE Tran. Advanced Packaging*, vol. 28, pp. 754–760, Nov. 2005.

[33] Chen, G. and Friedman, E. G., "Effective capacitance of RLC loads for estimating short-circuit power," in *Proc. IEEE Int. Symp. Circuits and Systems*, pp. 21–24, May 2006.

[34] Chen, J. and He, L., "Efficient in-package decoupling capacitor optimization for I/O power integrity," *IEEE Tran. Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, pp. 734–73, Apr. 2007.

[35] Chen, T. and Naffziger, S., "Comparison of adaptive body bias (ABB) and adaptive supply voltage (ASV) for improving delay and leakage under the presence of process variation," *IEEE Tran. Very Large Scale Integration (VLSI) Systems*, vol. 11, pp. 888–899, Oct. 2003.

[36] Chi, J. C., Huang, T. H., and Chi, M. C., "An IR drop-driven placer for standard cells in a SOC design," in *Proc. IEEE Int. SOC Conf.*, pp. 29–32, Sept. 2005.

[37] Chiou, L.-Y., Muhammand, K., and Roy, K., "DSP data path synthesis for low-power applications," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1165–1168, May 2001.

[38] Chu, P. P. and Jones, R. E., "Design techniques of fpga based random number generator," in *Military and Aerospace Applications of Programmable Devices and Technologies Conf.*, pp. 203–230, 1999.

[39] Clausius, R., "On the moving force of heat and the laws of heat which may be deduced therefrom," *Philosophical Magazine*, vol. 2, 1851.[Translation of the original German text "Uber die Bewegende Kraft der Warme" published in 1850 in Annalen der Physik].

[40] Connelly, J. A. and Choi, P., *Macromodeling with SPICE*. Englewood Cliffs, N.J.: Prentice Hall, 1992.

[41] Cormen, T., Leiserson, C., and Rivest, R., *Introduction to algorithms*. Cambridge, Mass.: MIT Press and McGraw-Hill, 2001.

[42] Coudert, O., "Gate sizing for constrained delay/power/area optimization," *IEEE Tran. Very Large Scale Integration (VLSI) Systems*, vol. 5, pp. 465–472, Dec. 1997.

[43] Cover, T. M. and Thomas, J. A., *Elements of Information Theory*. New York: Wiley, 1991.

[44] Daasch, W. R., Lim, C. H., and Cai, G., "Design of VLSI CMOS circuits under thermal constraint," *IEEE Tran. Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 49, pp. 589–593, Aug. 2002.

[45] Ding, Y. Z. and Rabin, M. O., "Hyper-encryption and everlasting security," *Lecture Notes In Computer Science; Proceedings of the 19th Annual Symposium on Theoretical Aspects of Computer Science*, vol. 2285, pp. 1–26, 2002.

[46] Doob, J. L., "The development of rigor in mathematical probability (1900-1950)," *American Mathematical Monthly*, vol. 103, pp. 586–595, Aug.-Sep. 1996.

[47] Elgamel, M. A. and Bayoumi, M. A., "Interconnect noise analysis and optimization in deep submicron technology," *IEEE Circuits and Systems Magazine*, vol. 3, pp. 6–17, 2003.

[48] Elias, P., "Computation in the presence of noise," *IBM Journal*, vol. 2, pp. 346–352, Oct. 1958.

[49] Ermolova, N. and Haggman, S., "Simplified bounds for the complementary error function; application to the performance evaluation of signal-processing systems," in *Proc. 12th European Signal Processing Conf.*, pp. 1087–1090, Sept. 2004.

[50] Fairfield, R., Mortenson, R. L., and Coulthart, K. B., "An lsi random number generator (rng)," in *Advances in Cryptology – CRYPTO '84*, pp. 203–230, 1984.

[51] Ferrenberg, A. M., Landau, D. P., and Wong, Y. J., "Monte carlo simulations: Hidden errors from "good" random number generators," *Phys. Rev. Let*, vol. 69, pp. 3382–3384, 1992.

[52] Fetzer, E. S., "Using adaptive circuits to mitigate process variations in a microprocessor design," *IEEE Design and Test of Computers*, vol. 23, pp. 476–483, Nov. 2006.

[53] Fuks, H., "Non-deterministic density classification with diffusive probabilistic cellular automata," *Physical Review E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 66, no. 066106, 2002.

[54] Gelenbe, E. and Batty, F., "Minimum graph covering with the random neural network model," in *Neural Networks: Advances and Applications*, vol. 2, 1992.

[55] George, J., Marr, B., Akgul, B., and Palem, K., "Probabilistic arithmetic and energy efficient embedded signal processing," in *Proc. Int. Conf. Cmpilers, Architectures and Synthesis for Embedded Systems*, pp. 158–168, Oct. 2006.

[56] Gibbs, J., *Elementary Principles in Statistical Mechanics*. New York: Scribner, 1902.

[57] Gikhman, I. I. and Skorokhod, A. V., *Introduction to the Theory of Random Processes*. New York: Dover Publications, 1996.

[58] Goel, S., Elgamel, M., Bayoumi, M., and Hanafy, Y., "Design methodologies for high-performance noise-tolerant XOR-XNOR circuits," *IEEE Tran. Circuits and Systems I*, vol. 53, pp. 867–878, Apr. 2006.

[59] Goel, S., Kumar, A., and Bayoumi, M. A., "Design of robust, energy-efficient full adders for deep-submicrometer design using hybrid-CMOS logic style," *IEEE Trans. Very Large Scale Integration (VLSI)*, vol. 14, pp. 1309–1321, Dec. 2006.

[60] Gonzalez, R., Gordon, B. M., and Horowitz, M. A., "Supply and threshold voltage scaling for low power CMOS," *IEEE Journal of Solid-State Circuits*, vol. 32, pp. 1210–1216, Aug. 1997.

[61] Goodman, J. and Chandrakasan, A. P., "Low power scalable encryption for wireless systems," *Wireless Networks*, vol. 4, pp. 55–7, Jan. 1998.

[62] Hajek, B. and Weller, T., "On the maximum tolerable noise for reliable computation by formulas," *IEEE Trans. Inform. Theory*, vol. 37, pp. 388–391, Mar. 1991.

[63] Halmos, P. R., *Measure Theory*. New York: Springer-Verlag, 1974.

[64] Halter, J. and Najm, F., "A gate-level leakage power reduction method for ultra-low-power CMOS circuits," in *Proc. IEEE Custom Integrated Circuits Conf.*, pp. 475–478, May 1997.

[65] Hamoui, A. A. and Rumin, N. C., "An analytical model for current, delay, and power analysis of submicron CMOS logic circuits," *IEEE Tran. Circuits and Systems II: Analog and Digital Signal Processing*, vol. 47, pp. 999–1007, Oct. 2000.

[66] Hartmanis, J. and Stearns, R. E., "On the computational complexity of algorithms," *Transactions of the American Mathematical Society*, vol. 117, pp. 285–306, May 1965.

[67] Hawkins, T., *Lebesgue's Theory of Integration: Its Origins and Developments*. New York: Chelsea Pub. Co., 1975.

[68] Heath, J. R., Kuekes, P. J., Snider, G. S., and Williams, R. S., "A defect-tolerant computer architecture: Opportunities for nanotechnology," *Science*, vol. 280, pp. 1716–1721, June 1998.

[69] Hedenstierna, N. and Jeppson, K. O., "CMOS circuit speed and buffer optimization," *IEEE Tran. Computer-Aided Design of Integrated Circuits and Systems*, vol. 6, pp. 270–281, Mar. 1987.

[70] Hegde, R. and Shanbhag, N. R., "Toward achieving energy efficiency in presence of deep submicron noise," *IEEE Trans. VLSI Syst.*, vol. 8, pp. 379–391, Aug. 2000.

[71] Henzler, S., Georgakos, G., Berthold, J., and Eireiner, M., "Activation technique for sleep-transistor circuits for reduced power supply noise," in *Proc. European Solid-State Circuits Conf. (ESSCIRC)*, pp. 102–105, Sept. 2006.

[72] Hirata, A., Onodera, H., and Tamaru, K., "Estimation of short-circuit power dissipation for static CMOS gates," *IEICE Transactions on Fundamental of Electronics, Communications and Computer Sciences*, vol. E79A, pp. 304–311, Mar. 1996.

[73] Ho, R., Mai, K., Kapadia, H., and Horowitz, M., "Interconnect scaling implications for cad," in *Proc. Int. Conf. Computer Aided Design*, pp. 425–429, Nov. 1999.

[74] Hofstee, H. P., "Power-constrained microprocessor design," in *Proc. Int. Conf. on Computer Design: VLSI in Computers and Processess*, (Freiburg, Germany), Sept. 2002.

[75] Holman, W. T., Connelly, J. A., and Dowlatabadi, A. B., "An integrated analog/digital random noise source," *IEEE Tran. Circuits and Systems I*, vol. 44, pp. 521–528, June 1197.

[76] Hooge, F. N., "1/f noise sources," *IEEE Trans. on Electron Devices*, vol. 41, pp. 1926–1935, Nov. 1994.

[77] Hrishikesh, M., Jouppi, N. P., Farkas, K. I., Burger, D., Keckler, S. W., and Shivakumar, P., "The optimal logic depth per pipeline stage is 6 to 8 fo4 inverter delays," in *Proc. Annual Int. Symp. Computer Architecture*, pp. 14–24, May 2002.

[78] HSPICE. http://www.synopsys.com/products/mixedsignal/hspice/hspice.html, Jan. 2003.

[79] Hung, K. K., Ko, P. K., Hu, C., and Cheng, Y. C., "A unified model for the flicker noise in metal-oxide- semiconductor field-effect transistors," *IEEE Trans. on Electron Devices*, vol. 37, pp. 654–665, Mar. 1990.

[80] IIT, "The OSU standard cell library." `[Online]. Available: http://www.ece.iit.edu/~jgrad/osucells/`, May 2005.

[81] Iman, S. and Pedram, M., "Two-level logic minimization for low power," in *IEEE/ACM Computer-Aided Design (ICCAD)*, pp. 433–438, Nov. 1995.

[82] ITRS 2005 Edition, "http://www.itrs.net/links/2005itrs/home2005.htm," Feb. 2006.

[83] Jakobson, C., Bloom, I., and Nemirovsky, Y., "1/f noise in CMOS transistors for analog applications from subthreshod to saturation," *Solid-State Electronics*, vol. 42, pp. 1807–1817, 1998.

[84] Jaynes, E. T. and Bretthorst, G. L., *Probability Theory: The Logic of Science*. Cambridge, United Kingdom: Cambridge University Press, 2003.

[85] Jeong, W. and Roy, K., "High-performance low-power dual transition preferentially sized (DTPS) logic," *IEEE J. Solid-State Circuits*, vol. 40, no. 2, pp. 480–484, 2005.

[86] Jessa, M., "Designing security for number sequences generated by means of the sawtooth chaotic map," *IEEE Tran. Circuits and Systems I*, vol. 53, pp. 1140–1150, May 2006.

[87] Johnson, J. B., "Thermal agitation of electrical charge in conductors," *Physical Review*, vol. 32, pp. 97–109, July 1928.

[88] Johnson, M., Somasekhar, D., Chiou, L., and Roy, K., "Leakage control with efficient use of transistor stacks in single threshold CMOS," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 10, pp. 1–5, Feb. 2002.

[89] J.Walker, "Hotbits: Genuine random numbers, generated by radioactive decay." `[Online]. Available: http://www.fourmilab.ch/hotbits/`, July 2004.

[90] Kallenberg, O., *Foundations of Modern Probability*. New York: Springer-Verlag, 2001.

[91] Kao, J., Narendra, S., and Chandrakasan, A., "Subthreshold leakage modeling and reduction techniques [IC CAD tools]," in *IEEE/ACM Int. Conf. Computer-Aided Design (ICCAD)*, pp. 141–148, Nov. 2002.

[92] Karp, R. M., "An introduction to randomized algorithms," *Discrete Applied Mathematics*, vol. 34, pp. 165–201, 1991.

[93] Keshavarzi, A., Ma, S., Narendra, S., Bloechel, B., Mistry, K., Ghani, T., Borkar, S., and De, V., "Effectiveness of reverse body bias for leakage control in scaled dual v_t CMOS ICs," in *Proc. Int. Symp. Low Power Electronics and Design*, pp. 207–212, Aug. 2001.

[94] Khan, Z., Arslan, T., and Erdogan, A. T., "Low power system on chip bus encoding scheme with crosstalk noise reduction capability," *IEE Proceedings Computers and Digital Techniques*, vol. 153, pp. 101–108, Mar. 2006.

[95] Khandelwal, V. and Srivastava, A., "A general framework for accurate statistical timing analysis considering correlations," in *Proc. Design Automation Conf. (DAC) 2005*, pp. 89–94, June 2005.

[96] Kim, C. H. and Roy, K., "Dynamic V_{th} scaling scheme for active leakage power reduction," in *Proc. Design Automation and Test in Europe (DATE)*, pp. 163–167, Mar. 2002.

[97] Kim, J., Yang, J., and Hwang, S.-Y., "Path sensitisation and gate sizing approach to low power optimisation," *Electronics Letters*, vol. 34, pp. 619–620, Apr. 1998.

[98] Kim, P. and Lieber, C. M., "Nanotube nanotweezers," *Science*, vol. 286, pp. 2148–2150, Dec. 1999.

[99] Kirton, M. J. and Uren, M. J., "Noise in solid-state microstructures: A new perspective on individual defects, interface states and low-frequency (1/f) noise," *Advances in Physics*, vol. 38, pp. 367–468, 1989.

[100] Kish, L. B., "End of Moore's law: thermal (noise) death of integration in micro and nano electronics," *Physics Letters A*, vol. 305, pp. 144–149, Dec. 2002.

[101] Ko, U. and Balsara, P. T., "Short-circuit power driven gate sizing technique for reducing power dissipation," *IEEE Tran. Very Large Scale Integration (VLSI) Systems*, vol. 3, pp. 450–455, Sept. 1995.

[102] Kolmogoroff, A. N., *Grundbegriffe der Wahrscheinlichkeitsrechnung.* Berlin: Springer, 1933. [English Translated Edition: A. N. Kolmogorov, *Foundations of the Theory of Probability*, New York: Chelsea Pub. Co., 1950].

[103] Korkmaz, P., Akgul, B. E. S., and Palem, K. V., "Characterizing the behavior of a probabilistic CMOS switch through analytical models and its verification through simulations," Tech. Rep. CREST-TR-05-08-01, Available at http://www.crest.gatech.edu/palempbitscurrent/, Aug. 2005.

[104] Kuroda, T., Suzuki, K., Mita, S., Fujita, T., Ymane, F., Sano, F., Chiba, A., Watanabe, Y., Matsuda, K., Maeda, T., Sakurai, T., and Furuyama, T., "Variable supply-voltage scheme for low-power high-speed CMOS digital design," *IEEE J. of Solid-State Circuits*, vol. 33, pp. 454–462, Mar. 1998.

[105] Landauer, R., "Irreversibility and heat generation in the computing process," *IBM J. Research and Development*, vol. 3, pp. 183–191, July 1961.

[106] Lee, D., Blaauw, D., and Sylvester, D., "Static leakage reduction through simultaneous V_t/t_{ox} and state assignment," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1014–1029, July 2005.

[107] Lee, D., Kwong, W., Blaauw, D., and Sylvester, D., "Analysis and minimization techniques for total leakage considering gate oxide leakage," in *Proc. Design Automation Conf. (DAC)*, pp. 175–180, June 2003.

[108] Li, H., Fan, J., Tan, S. X. D., Wu, L., Cai, Y., and Hong, X., "Partitioning-based approach to fast on-chip decoupling capacitor budgeting and minimization," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, pp. 2402–2412, Nov. 2006.

[109] Li, X. R., *Probability, Random Signals, and Statistics.* Boca Raton: CRC Press LLC, 1999.

[110] Likharev, K. K., "Single-electron devices and their applications," *Proceedings of the IEEE*, vol. 87, pp. 606–632, Apr. 1999.

[111] Liu, M., Wang, W.-S., and Orshansky, M., "Leakage power reduction by dual-Vth designs under probabilistic analysis of Vth variation," in *Proc. Int. Symp. Low Power Electronics and Design (ISLPED)*, pp. 2–7, Aug. 2004.

[112] Lyonnard, D., Yoo, S., Baghdadi, A., and Jerraya, A. A., "Automatic generation of application-specific architectures for heterogeneous multiprocessor system-on-chip," *Proc. Design Automation Conf. (DAC)*, pp. 518–523, 2001.

[113] MacKay, D. J. C., "Bayesian interpolation," *Neural Computation*, vol. 4, pp. 415–447, May 1992.

[114] Madsen, A. L. and Jensen, F. V., "Lazy propagation: a junction tree inference algorithm based on lazy evaluation," *Articial Intelligence*, vol. 113, pp. 203–245, 1999.

[115] Marculescu, D. and Talpes, E., "Energy awareness and uncertainty in microarchitecture-level design," *IEEE Micro*, vol. 25, no. 5, pp. 64–76, 2005.

[116] Markovic, D., Stojanovic, V., Nikolic, B., Horowitz, M. A., and Brodersen, R. W., "Methods for true energy-performance optimization," *IEEE J. Solid-State Circuits*, vol. 39, no. 8, pp. 1282–1293, 2004.

[117] Markovic, D., Stojanovic, V., Nikolic, B., Horowitz, M. A., and Brodersen, R. W., "Methods for true energy-performance optimization," *IEEE J. of Solid-State Circuits*, vol. 39, no. 8, pp. 1282–1292, 2004.

[118] Martin, S. M., Flautner, K., Mudge, T., and Blaauw, D., "Combined dynamic voltage scaling and adaptive body biasing for lower power microprocessors under dynamic workloads," in *IEEE/ACM Int. Conf. Computer Aided Design (ICCAD)*, pp. 721–725, Nov. 2002.

[119] Mathyas, S. M. and Meyer, C. H., "Generation, distribution and installation of cryptographic keys," *IBM Syst. J.*, vol. 17, pp. 126–137, 1978.

[120] Maxwell, J., "Illustrations of the dynamical theory of gases," *Philosophical Magazine*, vol. 19, pp. 19–32, 1860.

[121] Maxwell, J., "On the dynamical theory of gases," *Philosophical Transactions of the Royal Society of London*, vol. 157, pp. 49–88, 1867.

[122] McWhorter, A. L., "1/f noise and related surface effects in germanium," Tech. Rep. 80, Lincoln Lab, Boston, MA, May 1955.

[123] Meindl, J. D., "Low power microelectronics: Retrospect and prospect," *Proceedings of IEEE*, pp. 619–635, Apr. 1995.

[124] Meindl, J. D., "Low power microelectronics: retrospect and prospect," *Proc. IEEE*, vol. 83, pp. 619–635, Apr. 1995.

[125] Meindl, J. D. and Davis, J. A., "The fundamental limit on binary switching energy for terascale integration (tsi)," *IEEE. J. Solid-State Circuits*, pp. 1515–1516, Oct. 2000.

[126] Meindl, J. D., Chen, Q., and Davis, J. A., "Limits on silicon nanoelectronics for terascale integration," *Science*, vol. 293, pp. 2044–2049, Sept. 2001.

[127] Mendoza-Hernandez, F., Linares-Aranda, M., and Champac-Vilela, V. H., "The noise immunity of dynamic digital circuits with technology scaling," in *Proc. of Int. Symp. Circuits and Systems (ISCAS)*, (Vancouver, Canada), pp. 493–496, May 2004.

[128] Mezhiba, A. V. and Friedman, E. G., "Scaling trends of on-chip power distribution noise," *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, vol. 12, pp. 386–394, Apr. 2004.

[129] Middleton, D., *An Introduction to Statistical Communication Theory*. New Jersey: Wiley-IEEE Press, 1996.

[130] Milter, O. and Kolodny, A., "Crosstalk noise reduction in synthesized digital logic circuits," *IEEE Tran. Very Large Scale Integration (VLSI) Systems*, vol. 11, pp. 1153–1158, Dec. 2003.

[131] Moivre, A. D., *The Doctrine of Chances: or, A Method of Calculating Probabilities of Events in Play*. London: Pearson, 1718. [3rd Reprinted Edition: New York: Chelsea Pub. Co., Apr. 2000].

[132] Moore, G. E., "Cramming more components onto integrated circuits," *Electronics Magazine*, pp. 114–117, Apr. 1965.

[133] MOSIS, "http://www.mosis.org," Jan. 2004.

[134] Motchenbacher, C. D. and Connelly, J. A., *Low-noise electronic system design*. New York, NY: John Wiley & Sons, Inc., 1993.

[135] Motwani, R. and Raghavan, P., *Randomized Algorithms*. Cambridge University Press, 1995.

[136] Mui, M., Banerjee, K., and Mehrotra, A., "Supply and power optimization in leakage-dominant technologies," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, pp. 1362–1371, Sept. 2005.

[137] Mukherjee, A., "On the reduction of simultaneous switching in SoCs," in *Proc. IEEE Computer Society Annual Symp. VLSI (ISVLSI)*, pp. 262–263, Feb. 2004.

[138] Mukhopadhyay, S., Bhunia, S., and Roy, K., "Modeling and analysis of loading effect in leakage of nano-scaled bulk-CMOS logic circuits," in *Proc. Design Automation and Test in Europe (DATE)*, pp. 224–229, Mar. 2005.

[139] Mukhopadhyay, S., Bhunia, S., and Roy, K., "Modeling and analysis of loading effect on leakage of nanoscaled bulk-CMOS logic circuits," *IEEE Tran. Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, pp. 1486–1495, Aug. 2006.

[140] Mukhopadhyay, S., Raychowdhury, A., and Roy, K., "Accurate estimation of total leakage current in scaled CMOS logic circuits based on compact current modeling," in *Proc. Design Automation Conf. (DAC)*, pp. 169–174, June 2003.

[141] Murry, H. F., "A general approach for generating natural random variables," *IEEE Trans. Computers*, vol. 19, pp. 1210–1213, 1970.

[142] Mutlu, O., Hyesoon, K., and Patt, Y. N., "Efficient runahead execution: Power-efficient memory latency tolerance," *IEEE Micro*, vol. 26, no. 1, pp. 10–20, 2006.

[143] Mutoh, S., Douseki, T., Matsuya, Y., Aoki, T., Shigematsu, S., and Yamada, J., "1-V power supply high-speed digital circuit technology with multi-threshold voltage CMOS," *IEEE J. Solid-State Circuits*, vol. 30, pp. 847–854, Aug. 1995.

[144] Na, N., Budell, T., Chiu, C., Tremble, E., and Wemple, I., "The effects of on-chip and package decoupling capacitors and an efficient ASIC decoupling methodology," in *Proc. Electronic Components and Technology Conf.*, pp. 556–567, June 2004.

[145] Narendra, S., Blaauw, D., Devgan, A., and Najm, F., "Leakage issues in IC design: trends, estimation, and avoidance," in *Proc. of Int. Conf. on Computer Aided Design (ICCAD)*, Nov. 2003.

[146] Narendra, S., De, V., Borkar, S., Antoniadis, D., and Chandrakasan, A., "Full-chip sub-threshold leakage power prediction model for sub-0.18 μm CMOS," in *Proc. Int. Low Power Electronics and Design (ISLPED)*, pp. 19–23, May 2002.

[147] Narendra, S., Keshavarzi, A., Bloechel, B. A., Borkar, S., and De, V., "Forward body bias for microproessors in 130-nm technology generation and beyond," *IEEE Journal of Solid-State Circuits*, vol. 38, pp. 696–701, May 2003.

[148] Naroska, E., Ruan, S.-J., and Schwiegelshohn, U., "Simultaneously optimizing crosstalk and power for instruction bus coupling capacitance using wire pairing," *IEEE Tran. Very Large Scale Integration (VLSI) Systems*, vol. 14, pp. 421–425, Apr. 2006.

[149] Natori, K. and Sano, N., "Scaling limit of digital circuits due to thermal noise," *Journal of Applied Physics*, vol. 83, pp. 5019–5024, May 1998.

[150] Nepal, K., Bahar, R. I., Mundy, J., Patterson, W. R., and A.Zaslavsky, "Designing logic circuits for probabilistic computation in the presence of noise," in *Proc. Design Automation Conf. (DAC)*, pp. 485–490, June 2005.

[151] Nose, K. and Sakurai, T., "Analysis and future trend of short-circuit power," *IEEE Tran. Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, pp. 1023–1030, Sept. 2000.

[152] Nyquist, H., "Thermal agitation of electrical charge in conductors," *Physical Review*, vol. 32, pp. 110–113, July 1928.

[153] Packan, P. A., "Pushing the limits," *Science*, vol. 285, pp. 2079–2081, Sept. 1999.

[154] Palem, K. V., "Proof as experiment:probabilistic algorithms from a thermodynamic perspective," in *Proc. Int. Symposium on Verification (Theory and Practice)*, pp. 524–547, June 2003.

[155] Palem, K. V., "Energy aware computing through probabilistic switching: A study of limits," *IEEE Trans. Computer*, vol. 54, pp. 1123–1137, Sept. 2005.

[156] Palem, K. V., Akgul, B. E. S., , and George, J., "Variable scaling for computing elements," *OIT Invention Disclosure*, Feb. 2006.

[157] Pant, S., Blaauw, D., Zolotov, V., Sundareswaran, S., and Panda, R., "A stochastic approach to power grid analysis," in *Proc. Design Automation Conf. (DAC)*, (San Diego, California), pp. 171 – 176, June 2004.

[158] Papoulis, A. and Pillai, S. U., *Probability, Random Variables and Stochastic Processes*. New York: McGraw-Hill Publishing Co., 2002.

[159] Park, J.-K., Deshpande, H. V., and Woo, J. C. S., "Enhanced subthreshold leakage current due to impact ionization in deep sub-100nm n-channel double-gate MOSFETs," in *IEEE Int. SOI Conf.*, pp. 147–148, Oct. 2001.

[160] Park, S. and Miller, K. W., "Random number generators: good ones are hard to find," *Communications of the ACM*, vol. 31, Oct. 1988.

[161] Pavlovic, V., Garg, A., Rehg, J. M., and Huang, T. S., "Multimodal speaker detection using error feedback dynamic bayesian networks," in *Computer Vision and Pattern Recognition*, vol. 2, (Ft. Collins, CO), pp. 34–41, June 2000.

[162] Pelloie, J. L., "Using SOI to achieve low-power consumption in digital," in *Proc. SOI Conf.*, pp. 14–17, Oct. 2005.

[163] Penzes, P. I. and Martin, A. J., "Energy-delay efficiency of VLSI computations," in *Proc. GLSVLSI 2002*, (New York, USA), Apr. 2002.

[164] Petrie, C. S. and Connelly, J. A., "A noise-based IC random number generator for applications in cryptography," *IEEE Transactions on Circuits and Systems, Part I*, vol. 47, pp. 615–621, Dec. 2000.

[165] Pfeffer, A., *Probabilistic Reasoning for Complex Systems.* PhD thesis, Stanford University, 2000.

[166] Pippenger, N., "Reliable computation by formulas in the presence of noise," *IEEE Trans. Inform. Theory*, vol. 34, pp. 194–197, Mar. 1988.

[167] Planck, M., *Treatise on Thermodynamics.* New York: Dover Publications, 1990.

[168] Qi, X., Lo, S. C., Gyure, A., Yansheng, L., Shahram, M., Singhal, K., and Macmillen, D. B., "Efficient subthreshold leakage current optimization - leakage current optimization and layout migration for 90- and 65- nm ASIC libraries," *IEEE Circuits and Devices Magazine*, vol. 22, pp. 39–47, Sept.- Oct. 2006.

[169] Queisser, H. J. and Haller, E. E., "Defects in semiconductors: Some fatal, some vital," *Science*, vol. 281, pp. 945–950, Aug. 1998.

[170] Rabin, M. O., "Probabilistic algorithms, Algorithms and Complexity, New Directions and Recent Trends (ed. J. F. Traub)," pp. 21–39, 1976.

[171] Rabin, M. O., "Complexity of computations," *Communications of the ACM*, vol. 20, pp. 625–633, Sept. 1977.

[172] Raghunathan, A. and Jha, N. K., "An iterative improvement algorithm for low power data path synthesis," in *IEEE/ACM Computer-Aided Design (ICCAD)*, pp. 597–602, Nov. 1995.

[173] Rahman, H. and Chakrabarti, C., "A leakage estimation and reduction technique for scaled CMOS logic circuits considering gate leakage," in *Proc. Int. Symp. on Circuits and Systems (ISCAS)*, pp. 297–300, May 2004.

[174] Random Number Generation and Testing, "http://csrc.nist.gov/rng/," Aug. 2005.

[175] Rao, R. M., Burns, J. L., and Brown, R. B., "Circuit techniques for gate and sub-threshold leakage minimization in future CMOS technologies," in *Proc. European Solid-State Circuits Conf.*, pp. 313–316, Sept. 2003.

[176] Rao, R. M., Burns, J. L., Devgan, A., and Brown, R. B., "Efficient techniques for gate leakage estimation," in *Proc. Int. Low Power Electronics and Design (ISLPED)*, pp. 100–103, Aug. 2003.

[177] Rastogi, A., Chen, W., Sanyal, A., and Kundu, S., "An efficient technique for leakage current estimation in sub 65nm scaled CMOS circuits based on loading effect," in *Int. Conf. VLSI Design*, pp. 583–588, Jan. 2007.

[178] Rehg, J. M., Murphy, K. P., and Fieguth, P. W., "Vision-based speaker detection using bayesian networks," in *Computer Vision and Pattern Recognition*, (Ft. Collins, CO), pp. 110–116, June 1999.

[179] Richards, M., *Fundamentals of Radar Signal Processing*. New York: McGraw Hill Publishing, 2005.

[180] Rossello, J. L. and Segura, J., "Charge-based analytical model for the evaluation of power consumption in submicron CMOS buffers," *IEEE Tran. Computer-Aided Design of Integrated Circuits and Systems*, vol. 21, pp. 433–448, Apr. 2002.

[181] Roy, K., Mukhopadhyay, S., and Mahmoodi-Meimand, H., "Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits," *Proceedings of the IEEE*, vol. 91, pp. 305–327, Feb. 2003.

[182] Sakurai, T. and Newton, A. R., "A simple short-channel MOSFET model and its application to delay analysis of inverters and series-connected MOSFETs," in *Proc. Int. Symp. on Circuits and Systems (ISCAS)*, pp. 105–108, May 1990.

[183] Sakurai, T. and Newton, A. R., "Delay analysis of series-connected MOSFET circuits," *IEEE J. of Solid-State Circuits*, vol. 26, pp. 122–131, Feb. 1991.

[184] Sano, N., "Increasing importance of electronic thermal noise in sub-0.1mm Si-MOSFETs," *IEICE Transactions on Electronics*, vol. E83-C, pp. 1203–1211, Aug. 2000.

[185] Schottky, W., "Über spontane stromschwankungen in verschiedenen elektrizitätsleitern," *Annalen der Physik*, vol. 57, pp. 541–567, 1918.

[186] Shannon, C. E., "A mathematical theory of communications," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 623–656, July, Oct. 1948.

[187] Shannon, C. E., "Communication theory of secrecy systems," *Bell Syst. Tech. J.*, vol. 28, pp. 656–715, 1949.

[188] Shen, S.-J., Lin, C.-J., and Hsu, C. C.-H., "Ultra fast write speed, long refresh time, low power F-N operated volatile memory cell with stacked nanocrystalline Si film," in *Proc. Int. Electron Devices Meeting*, pp. 515–518, Dec. 1996.

[189] Shepard, K. L., "Design methodologies for noise in digital integrated circuits," in *Proc. Design Automation Conf. (DAC)*, (San Francisco, California), pp. 94–99, June 1998.

[190] Shepard, K. L. and Narayanan, V., "Conquering noise in deep-submicron digital ICs," *IEEE Design and Test of Computers*, vol. 15, pp. 51–62, January-March 1998.

[191] Shichman, H. and Hodges, D., "Modeling and simulation of insulated-gate field-effect transistor switching circuits," *IEEE Journal of Solid-State Circuits*, vol. 3, pp. 285–289, Sept. 1968.

[192] Shim, B. and Shanbhag, N. R., "Energy-efficient soft error-tolerant digital signal processing," *IEEE Trans. VLSI Syst.*, vol. 14, pp. 336–348, Apr. 2006.

[193] Sirichotiyakul, S., T. Edwards, T., Oh, C., Zuo, J., Dharchoudhury, A., Panda, R., and Blaauw, D., "Stand-by power minimization through simultaneous threshold voltage selection and circuit sizing," in *Proc. Design Automation Conf. (DAC)*, pp. 436–441, June 1999.

[194] SMIPS32 M4K Processor Core Software Users Manual, "http://www.mips.com," July 2005.

[195] Solomatnikov, A., Somasekhar, D., Sirisantana, N., and Roy, K., "Skewed CMOS: noise tolerant high-performance low-power static circuit family," *IEEE Trans. VLSI Syst.*, vol. 10, no. 4, pp. 469–476, 2002.

[196] Solovay, R. and Strassen, V., "A fast monte-carlo test for primality," *SIAM Journal on Computing*, 1977.

[197] Stathis, J. and DiMaria, D., "Reliability projection for ultra-thin oxides at low voltage," in *Int. Electron Devices Meeting (IEDM)*, pp. 167–170, Dec. 1998.

[198] Stein, K.-U., "Noise-induced error rate as a limiting factor for energy per operation in digital ICs," *IEEE J. Solid-State Circuits*, vol. 12, pp. 527–530, Oct. 1977.

[199] Stojanovski, T. and Kocarev, L., "Chaos-based random number generators-part I: analysis [cryptography]," *IEEE Tran. Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, pp. 281–288, Mar. 2001.

[200] Stojanovski, T. and Kocarev, L., "Chaos-based random number generators. part ii: practical realization," *IEEE Tran. Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, pp. 382–385, Mar. 2001.

[201] Strecok, A. J., "On the calculation of the inverse of the error function," *Mathematics of Computation*, vol. 22, pp. 144–158, Jan. 1968.

[202] StrongARM-1100 Microprocessor Technical Reference Manual, "http://www.intel.com," July 2005.

[203] Sunar, B., Martin, W. J., and Stinson, D. R., "A provably secure true random number generator with built-in tolerance to active attacks," *IEEE Trans. Computers*, vol. 56, pp. 109–119, Jan. 2007.

[204] Sutherland, I., Sproull, B., and Harris, D., *Logical Effort: Designing Fast CMOS Circuits*. San Francisco, CA: Morgan Kauffmann Publishers, Inc., 1999.

[205] Swanson, R. and Meindl, J., "Ion-implanted complementary MOS transistors in low-voltage circuits," *IEEE Journal of Solid-State Circuits*, vol. 7, pp. 146–153, Apr. 1972.

[206] Szilard, L., "On the decrease of entropy in a thermodynamic system by the intervention of intelligent beings," *in Maxwell's Demon: Why warmth disperses and time passes by H. Leff and E. Rex*, 1998.

[207] Taur, Y. and Ning, T., *Fundamentals of modern VLSI devices*. New York, NY: Cambridge Univ. Press, 1998.

[208] Thompson, S., Packan, P., and Bohr, M., "Linear versus saturated drive current: Tradeoffs in super steep retrograde well engineering," in *Dig. Tech. Papers Symp. VLSI Technology*, pp. 154–155, June 1996.

[209] T.Rueckes, Kim, K., Joselevich, E., Tseng, G. Y., Cheung, C.-L., and Lieber, C. M., "Carbon nanotube-based nonvolatile random access memory," *Science*, vol. 289, pp. 94–97, July 2000.

[210] Tschanz, J. W., Kao, J. T., Narendra, S. G., Nair, R., Antoniadis, D. A., Chandrakasan, A. P., and De, V., "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage," *IEEE Journal of Solid-State Circuits*, vol. 37, pp. 1396–1402, Nov. 2002.

[211] Tseng, J.-M. and Jou, J.-Y., "A power driven two-level logic optimizer," in *Proc. Asia and South Pacific Design Automation Conf. (ASP-DAC)*, pp. 113–116, Jan. 1997.

[212] Tsividis, Y., *Operation and Modeling of the MOS Transistor*. Boston: McGraw-Hill, 1999.

[213] U.C. Berkeley BSIM Homepage, "http://www-device.eecs.berkeley.edu/bsim3/," Mar. 2005.

[214] Ulman, S., "Macromodel for short circuit power dissipation of submicron CMOS inverters and its application to design CMOS buffers," in *Proc. Int. Symp. Circuits and Systems (ISCAS)*, pp. 269–272, May 2003.

[215] Uyemura, J. P., *CMOS Logic Circuit Design*. Norwell, MA: Kluwer Academic Publishers, 2002.

[216] Van der Ziel, A., *Noise: Sources, Characterization, Measurement*. Englewood Cliffs: Prentice-Hall, 1970.

[217] Van Heijningen, M., Badaroglu, M., Donnay, S., Gielen, G. G. E., and Man, H. J. D., "Substrate noise generation in complex digital systems: Efficient modeling and simulation methodology and experimental verification," *IEEE J. of Solid-State Circuits*, vol. 37, pp. 1065–1072, Aug. 2002.

[218] Vandamme, L. K. J., Li, X., and Rigaud, D., "1/f noise in MOS devices, mobility or number fluctations?," *IEEE Trans. on Electron Devices*, vol. 41, pp. 1936–1944, Nov. 1994.

[219] Vazirani, U. V. and Vazirani, V. V., "Efficient and secure pseudo-random number generation (extended abstract)," in *Proc. Ann. Symp. Foundations of Computer Science*, pp. 458–463, 1984.

[220] Veendrick, H. J. M., "Short-circuit dissipation of static CMOS circuitry and its impact on the design of buffer circuits," *IEEE Journal of Solid-State Circuits*, vol. 19, pp. 468–473, Aug. 1984.

[221] Vemuru, S. R. and Scheinberg, N., "Short-circuit power dissipation estimation for CMOS logic gates," *IEEE Tran. Circuits and Systems I: Fundamental Theory and Applications*, vol. 41, pp. 762–765, Nov. 1994.

[222] Vincent, C. H., "The generation of truly random binary numbers," *Journal of Physics E*, vol. 3, pp. 594–598, 1970.

[223] Von Mises, R., *Wahrscheinlichkeitsrechnung, Statistik und Wahrheit.* Berlin: Springer, 1928. [2nd. rev. English Edition prepaed by Hilda Geiringer: R. von Mises, *Probability, Statistics and Truth*, Macmillan, 1957].

[224] Von Neumann, J., "Probabilistic logics and the synthesis of reliable organizms from unreliable components," in *Automata Studies* (Shannon, C. E. and McCarthy, J., eds.), (Princeton, NJ), pp. 43–98, Princeton Univ. Press, 1956.

[225] Von Neumann, J., *in Fourth University of Illinois lecture in Theory of Self-Reproducing Automata (A. W. Burks editor).* University of Illinois Press, 1966.

[226] VTVT, "VTVT standard cell library distribution." `[Online]. Available: http://www.ee.vt.edu/~ha/cell_library/distribution.html/`, May 2005.

[227] Wan, L., Raj, P. M., Balaraman, D., Muthana, P., Bhattarcharya, S. K., Varadarajan, M., Abothu, I. R., Sawminathan, M., and Tummala, R., "Embedded decoupling capacitor performance in high speed circuits," in *Proc. Electronic Components and Technology Conf.*, pp. 1617–1622, May-June 2005.

[228] Wang, A., Chandrakasan, A. P., and Konosocky, S. V., "Optimal supply and threshold voltage scaling for subthreshold CMOS circuits," in *Proc. IEEE Annual Int. Symp. on VLSI (ISVLSI '02)*, pp. 7–11, Apr. 2002.

[229] Wang, J.-C., Wang, J.-F., Suen, A.-N., and Weng, Y.-S., "A programmable application-specific VLSI architecture for speech recognition," *The 8th IEEE Int. Conf. on Electronics, Circuits and Systems (ICECS)*, vol. 1, pp. 477–480, 2001.

[230] Wang, Q. and Vrudhula, S. B. K., "On short-circuit power estimation of CMOS inverters," in *Proc. IEEE Int. Conf. on Computer Design*, pp. 70–75, Oct. 1998.

[231] Wenger, S., "H.264/AVC over IP," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, pp. 645–656, July 2003.

[232] Wong, E., Minz, J., and Lim, S. K., "Decoupling capacitor plannig and sizing for noise and leakage reduction," in *Proc. IEEE/ACM Int. Conf. Computer-Aided Design (ICCAD)*, pp. 395–400, Nov. 2006.

[233] Xtensa Microprocessor, "http://www.tensilica.com," July 2005.

[234] Yeh, W.-K. and Chou, J.-W., "Optimum halo structure for sub-0.1 μm CMOSFET's," *IEEE Trans. on Electron Devices*, vol. 48, pp. 2357–2362, Oct. 2001.

[235] Zhao, S., Koh, C.-K., and Roy, K., "Decoupling capacitance allocation and its application to power supply noise aware floorplanning," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, pp. 81–92, Jan. 2002.

[236] Zhirnov, V., Cavin, R. K., Hutchby, J. A., and Bourianoff, G. I., "Limits to binary logic switch scaling—a gedanken model," *Proceedings of the IEEE*, vol. 91, pp. 1934–1939, Nov. 2003.

[237] Zutic, I., Fabian, J., and Sarma, S. D., "Spintronics: Fundamentals and applications," *Reviews of Modern Physics*, vol. 76, pp. 323–410, Apr. 2004.

PUBLICATIONS

This dissertation is based on and/or related to the work and results presented in the following publications:

[1] P. Korkmaz, B. E. S. Akgul, and K. V. Palem, "Analysis of the Probability and Energy of Nanometer CMOS Circuits in the Presence of Noise", To appear in *Electronics Letters*, 2007.

[2] L. N. Chakrapani, P. Korkmaz, B. E. S. Akgul, and K. V. Palem, "Probabilistic System-on-a-chip Architectures", To appear in *ACM Transactions on Design Automation of Electronic Systems*, 2007.

[3] K. V. Palem, L. N. Chakrapani, B. E. S. Akgul, and P. Korkmaz, "A Review on Probabilistic CMOS (PCMOS) Technology: From Device Characteristics to Ultra Low-Energy SOC Architectures," in *High Performance Embedded Computing Handbook: A Systems Perspective*, edited by M. Michael Vai, David R. Martinez and Robert A. Bond, CRC Press LLC, 2007.

[4] B. E. S. Akgul, L. N. Chakrapani, P. Korkmaz, and K. V. Palem, "Probabilistic CMOS Technology: A Survey and Future Directions," in *Proc. IFIP Int. Conf. on Very Large Scale Integration*, pp. 1-6, Oct. 2006.

[5] P. Korkmaz, B. E. S. Akgul, K. V. Palem, and L. N. Chakrapani, "Advocating noise as an agent for ultra-low energy computing: probabilistic complementary metal-oxide-semiconductor devices and their characteristics," *Japanese Journal of Applied Physics*, Vol. 45, No. 4B, pp. 3307-3316, Apr. 2006.

[6] P. Korkmaz, B. E. S. Akgul, and K. V. Palem, "Ultra-low energy computing with noise: energy-performance-probability trade-offs," in *Proc. IEEE Computer Society Annual Symp. on VLSI*, pp. 349-354, Mar. 2006.

[7] L. N. Chakrapani, B. E. S. Akgul, S. Cheemalavagu, P. Korkmaz, K. V. Palem, and B. Seshasayee, "Ultra efficient embedded SOC architectures based on probabilistic CMOS (PCMOS) technology," in *Proc. Design Automation and Test in Europe (DATE)*, Mar. 2006.

[8] S. Cheemalavagu, P. Korkmaz, K. V. Palem, B. E. S. Akgul, and L. N. Chakrapani, "A Probabilistic Switch and its Realization by Exploiting Noise," in *Proc. IFIP Int. Conf. on Very Large Scale Integration (IFIP VLSI-SoC)*, pp. 452-457, Oct. 2005.

[9] K. V. Palem, L. N. Chakrapani, B. E. S. Akgul, and P. Korkmaz, "Realizing Ultra-low Energy Application Specific SoC Architectures through Novel Probabilistic CMOS (PCMOS) Technology," *Extended Abstracts of the 2005 Int. Conf. on Solid State Devices and Materials*, pp. 678-679, Sept. 2005.

[10] S. Cheemalavagu, P. Korkmaz, and K. V. Palem, "Ultra low-energy computing via probabilistic algorithms and devices: CMOS device primitives and the energy-probability relationship," *Proc. 2004 Int. Conf. on Solid State Devices and Materials*, pp. 402-403, Sept. 2004.

These are the publications of Pinar Korkmaz that are not related to this dissertation:

[1] M. Ekpanyapong, P. Korkmaz, and H. -H. S. Lee,, "Choice Predictor for Free," in *Proc. the 9th Asia-Pacific Computer Systems Architecture Conf.*, pp. 399-413, Sept. 2004.

[2] K. V. Palem, R. M. Rabbah, V. Mooney III, P. Korkmaz, and K. Puttaswamy, "Design Space Optimization of Embedded Memory Systems via Data Remapping," in *Proc. Languages, Software, Compilers and Tools for Embedded Systems (LCTES)*, pp. 28-37, June 2002.

[3] K. Puttaswamy, L. N. Chakrapani, K. W. Choi, Y. S. Dhillon, U. Diril, P. Korkmaz, K. K. Lee, J. C. Park, A. Chatterjee, P. Ellervee, V. Mooney, K. Palem, and W. F. Wong, "Power-Performance Trade-Offs in second level memory used by an ARM-Like RISC Architecture," in *Power Aware Computing, Ed. Rami Melhem, University of Pittsburgh, PA, USA and Robert Graybill, DARPA/ITO, Arlington, VA, USA. Kluwer Academic/Plenum Publishers*, pp. 211-224, May 2002.

[4] L. N. Chakrapani, P. Korkmaz, V. J. Mooney III, K. V. Palem, K. Puttaswamy, and W. -F. Wong, "The Emerging Power Crisis in Embedded Processors - What can a Poor Compiler Do?," in *Proc. Compilers Architecture and Synthesis of Embedded Systems*, pp. 176-181, Nov. 2001.

VITA

Pinar Korkmaz received a B.S. degree in Electrical and Electronics Engineering from Bogazici University, Istanbul in 1998, and an M.S. degree in Electrical and Electronics Engineering, also from Bogazici University, Istanbul in 2000. She joined the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta in August 2000, and received a Master of Science degree in Electrical and Computer Engineering from Georgia Institute of Technology, Atlanta in 2002. Her research towards PhD was done in the CREST lab under the guidance of Prof. Krishna V. Palem. Her areas of interest are digital VLSI circuit and system design, probabilistic circuits, and computer architectures.